Tomiwa Ademidun

Software Testing SE 4452: Assignment 2

1. Equivalent class Testing

AccountStatus

Class	Range (accountFactor)	Test Input Value
invalid	0	0
adverse	[1,99]	7
acceptable	[100,499]	300
good	[500,999]	700
excellent	[1000,∞]	1200

getAgeFactor

Class	Range (age)	Test Input Value
0	[-∞,14],[110,∞]	0, 120
5	[15,19]	17
10	[20,29]	20
20	[30,49],[65,109]	35, 77
50	[40,65]	55

getBalanceFactor

Class	Range (age)	Test Input Value
0	[-∞,0],[5000, ∞]	-100, 12000
6	[1,99]	17
16	[100,499]	200
30	[500,999]	550
70	[1000,2999]	1550
200	[3000,4999]	3500

Tomiwa Ademidun

creditStatus:

Class	Range (creditScore)	Test Input Value
Invalid	[-∞,0],[101, ∞]	-100, 120
Adverse, restricted	[1,49]	17
Adverse, default	[1,74]	200
Good, restricted	[50,100]	55
Good, default	[75,100]	80

productStatus:

Class	Range (productQuantity)	Test Input Value
soldout	0	0
limited	< inventoryThreshold	inventoryThreshold -10
available	>= inventoryThreshold	inventoryThreshold + 10

orderHandling:

Class	Range (productQuantity)	Test Input Value
accepted	excellent’ accountStatus ‘good’ accountStatus and creditStatus ‘adverse’ or ‘acceptable’ accountStatus, ‘good’ creditStatus and ‘available’ productStatus.	accountStatus = ‘excellent’
pending	acceptable’ accountStatus, ‘good’ creditStatus and ‘limited’ or ‘soldout’ productStatus.	accountStatus = acceptable’ creditStatus = ‘good’ productStatus = ‘soldout’
underReview	‘good’ accountStatus and ‘adverse’ creditStatus ‘acceptable’ accountStatus, ‘adverse’ creditStatus and ‘available’ productStatus	accountStatus = ‘good’ creditStatus = ‘adverse’ productStatus = ‘available’
rejected	‘acceptable’ accountStatus, ‘adverse’ creditStatus and ‘limited’ or ‘soldout’ productStatus. ‘adverse’	accountStatus = ‘adverse’ creditStatus = ‘good’ productStatus = ‘soldout’

	accountStatus, ‘good’ creditStatus and ‘soldout’ productStatus. ‘adverse’ accountStatus and ‘adverse’ creditStatus	

2. **Boundary Values Testing**

Account Status:

Class (result)	**Test Value(s)**
invalid	0
poor	1,99 (both not possible – no test case)
fair	100,499 (both not possible)
good	500,999 (both not possible)
Very good	2000 (both not possible)

getAgeFactor:

Class (result)	**Test Value(s)**
0	14,110
5	15,19
10	20,29
20	30,39
50	40,64

getBalanceFactor:

Class (result)	**Test Value(s)**
0	0,5000

6	1,99
16	100,499
30	500,999
70	1000,2999
200	3000,4999

creditStatus:

Class (result)	**Test Value(s)**
Invalid	-1, 101
Adverse, restricted	0,49
Adverse, default	0,74
Good, restricted	50,100
Good, default	75,100

productStatus:

Class (result)	**Test Value(s)**
soldout	0
limited	inventoryThreshold-1
available	inventoryThreshold+1

orderHandling: boundary test not applicable

3. Decision Table Testing

Tomiwa Ademidun

Blue	= Conditions
Green	= Actions

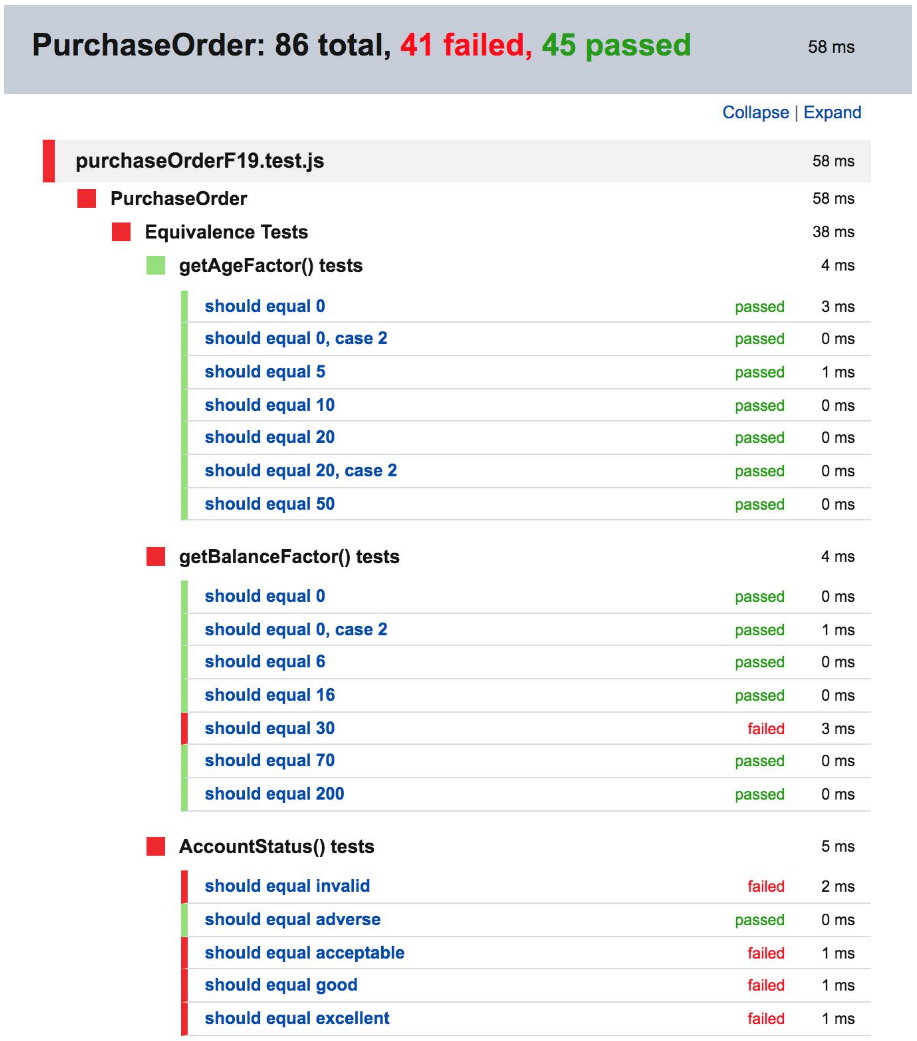
orderHandling:

	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7
accountStatus	excellent	good	adverse	acceptable	acceptable	acceptable	adverse
creditStatus	-	good	good	good	good	good	good
productStatus	-	-	available	available	limited	soldout	limited
accepted	X	X	X	X			
pending					X	X	X
underReview							
rejected							

	Rule 8	Rule 9	Rule 10	Rule 11	Rule 12	Rule 13	Rule 14
accountStatus	good	acceptable	invalid	acceptable	acceptable	adverse	adverse
creditStatus	adverse	adverse	invalid	adverse	adverse	good	adverse
productStatus	-	available	invalid	limited	soldout	soldout	-
accepted							
pending							
underReview	X	X					
rejected			X	X	X	X	X

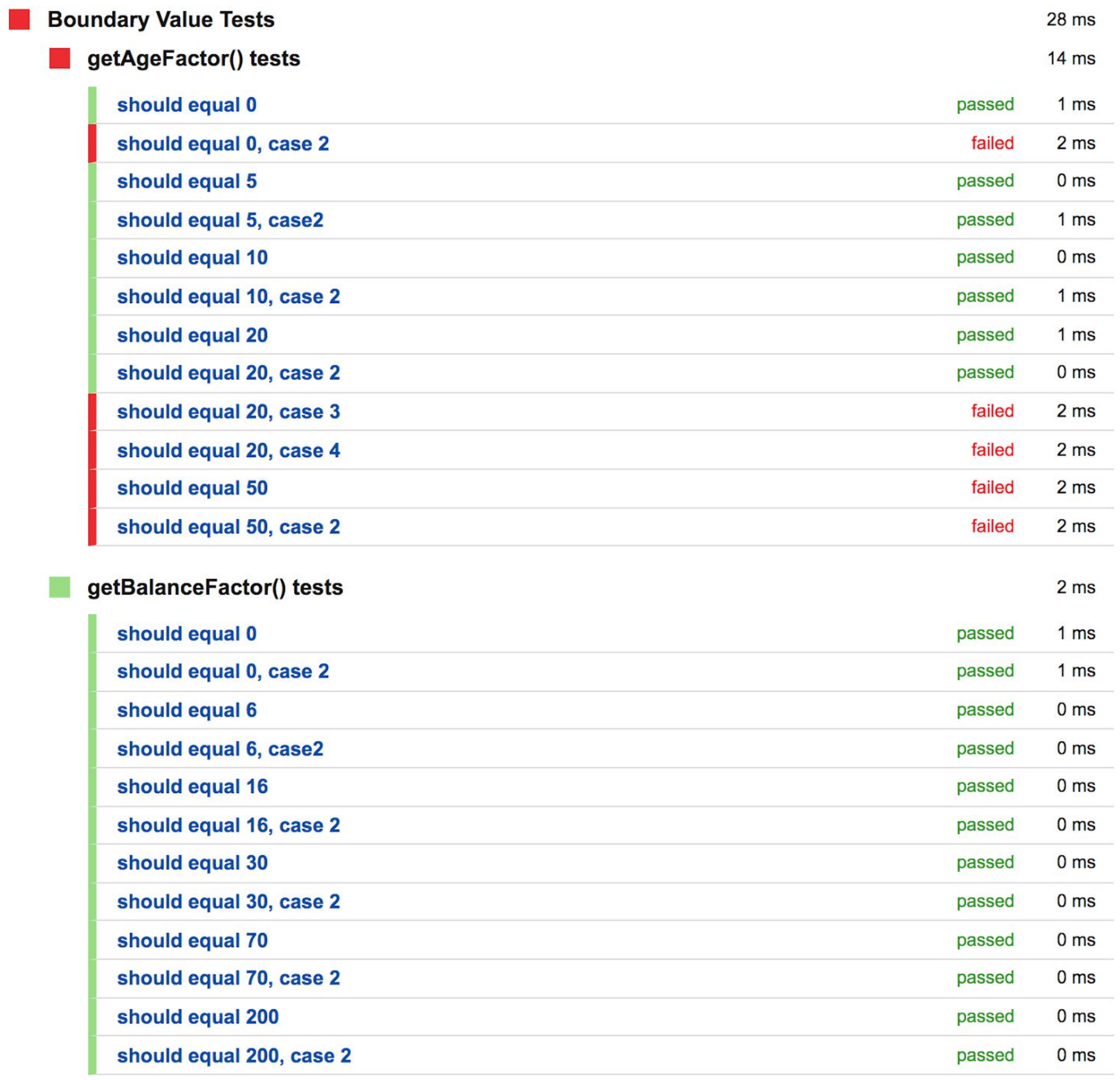
Tomiwa Ademidun

Analysis report:
Before Code Fixes



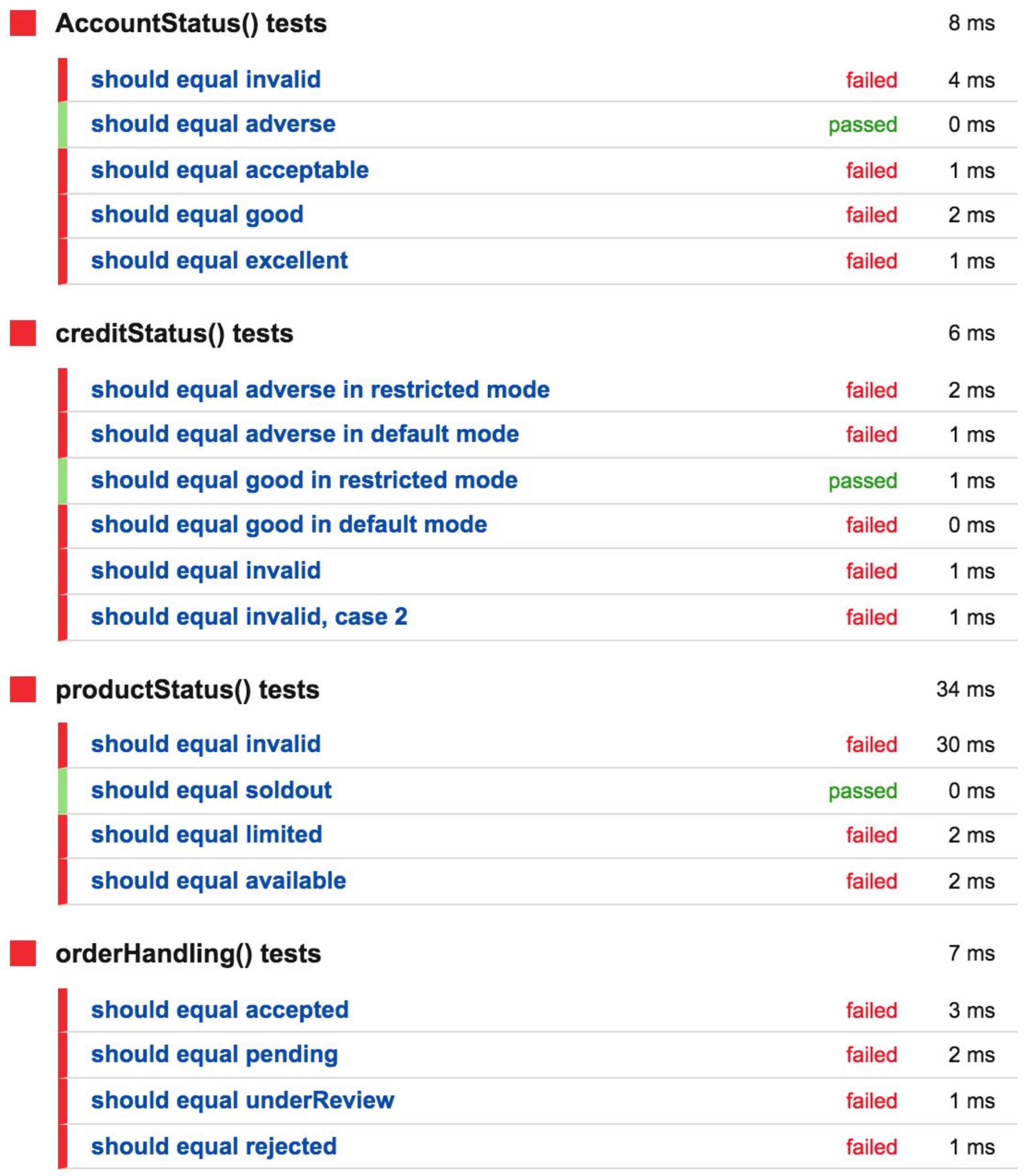
```
PurchaseOrder: 86 total, 41 failed, 45 passed                 58 ms

                                                   Collapse | Expand

purchaseOrderF19.test.js                                      58 ms
  PurchaseOrder                                               58 ms
    Equivalence Tests                                         38 ms
      getAgeFactor() tests                                     4 ms
        should equal 0                              passed     3 ms
        should equal 0, case 2                      passed     0 ms
        should equal 5                              passed     1 ms
        should equal 10                             passed     0 ms
        should equal 20                             passed     0 ms
        should equal 20, case 2                     passed     0 ms
        should equal 50                             passed     0 ms

      getBalanceFactor() tests                                 4 ms
        should equal 0                              passed     0 ms
        should equal 0, case 2                      passed     1 ms
        should equal 6                              passed     0 ms
        should equal 16                             passed     0 ms
        should equal 30                             failed     3 ms
        should equal 70                             passed     0 ms
        should equal 200                            passed     0 ms

      AccountStatus() tests                                    5 ms
        should equal invalid                        failed     2 ms
        should equal adverse                        passed     0 ms
        should equal acceptable                     failed     1 ms
        should equal good                           failed     1 ms
        should equal excellent                      failed     1 ms
```

Tomiwa Ademidun



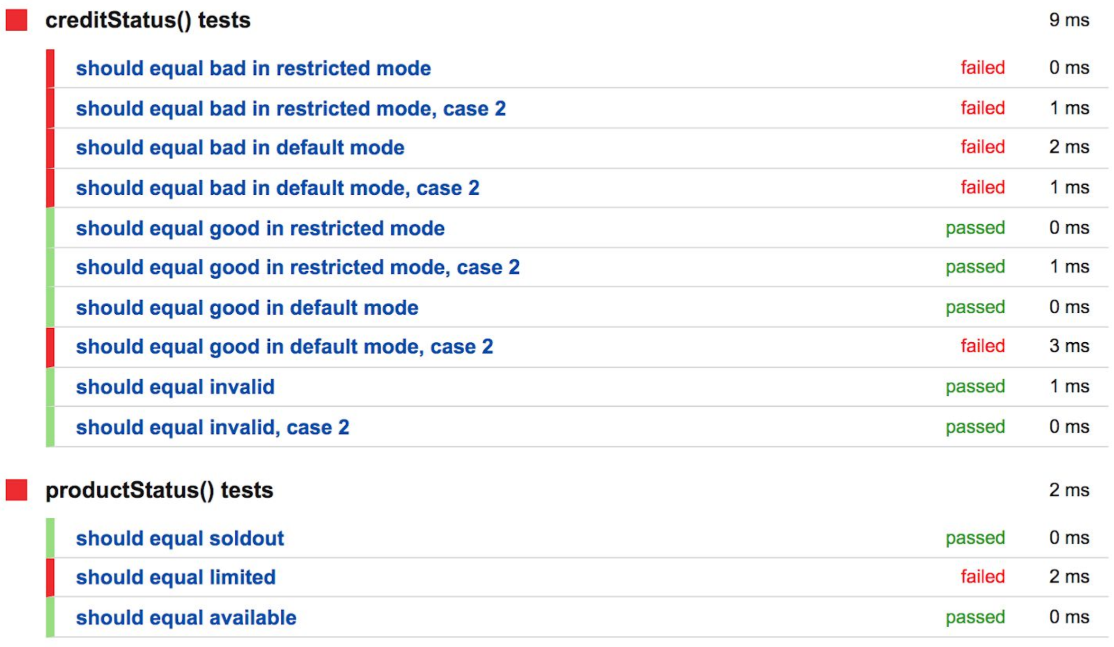
```
🟥 Boundary Value Tests                                              28 ms
  🟥 getAgeFactor() tests                                            14 ms
     should equal 0                                        passed     1 ms
     should equal 0, case 2                                failed     2 ms
     should equal 5                                        passed     0 ms
     should equal 5, case2                                 passed     1 ms
     should equal 10                                       passed     0 ms
     should equal 10, case 2                               passed     1 ms
     should equal 20                                       passed     1 ms
     should equal 20, case 2                               passed     0 ms
     should equal 20, case 3                               failed     2 ms
     should equal 20, case 4                               failed     2 ms
     should equal 50                                       failed     2 ms
     should equal 50, case 2                               failed     2 ms

  🟩 getBalanceFactor() tests                                         2 ms
     should equal 0                                        passed     1 ms
     should equal 0, case 2                                passed     1 ms
     should equal 6                                        passed     0 ms
     should equal 6, case2                                 passed     0 ms
     should equal 16                                       passed     0 ms
     should equal 16, case 2                               passed     0 ms
     should equal 30                                       passed     0 ms
     should equal 30, case 2                               passed     0 ms
     should equal 70                                       passed     0 ms
     should equal 70, case 2                               passed     0 ms
     should equal 200                                      passed     0 ms
     should equal 200, case 2                              passed     0 ms
```

Tomiwa Ademidun

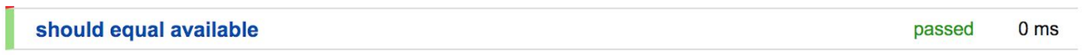


```
AccountStatus() tests                                          8 ms
    should equal invalid                              failed   4 ms
    should equal adverse                              passed   0 ms
    should equal acceptable                           failed   1 ms
    should equal good                                 failed   2 ms
    should equal excellent                            failed   1 ms

creditStatus() tests                                           6 ms
    should equal adverse in restricted mode           failed   2 ms
    should equal adverse in default mode              failed   1 ms
    should equal good in restricted mode              passed   1 ms
    should equal good in default mode                 failed   0 ms
    should equal invalid                              failed   1 ms
    should equal invalid, case 2                      failed   1 ms

productStatus() tests                                         34 ms
    should equal invalid                              failed  30 ms
    should equal soldout                              passed   0 ms
    should equal limited                              failed   2 ms
    should equal available                            failed   2 ms

orderHandling() tests                                          7 ms
    should equal accepted                             failed   3 ms
    should equal pending                              failed   2 ms
    should equal underReview                          failed   1 ms
    should equal rejected                             failed   1 ms
```

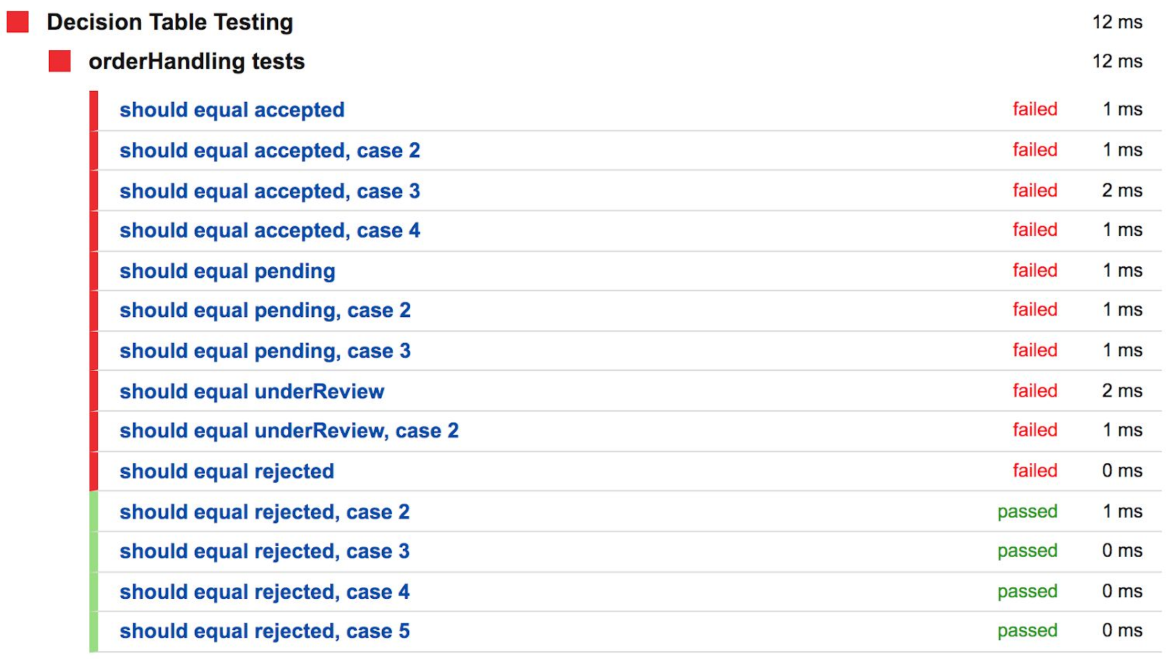
Tomiwa Ademidun



```
🟥 creditStatus() tests                                                   9 ms

   should equal bad in restricted mode                        failed     0 ms
   should equal bad in restricted mode, case 2                failed     1 ms
   should equal bad in default mode                           failed     2 ms
   should equal bad in default mode, case 2                   failed     1 ms
   should equal good in restricted mode                       passed     0 ms
   should equal good in restricted mode, case 2               passed     1 ms
   should equal good in default mode                          passed     0 ms
   should equal good in default mode, case 2                  failed     3 ms
   should equal invalid                                       passed     1 ms
   should equal invalid, case 2                               passed     0 ms

🟥 productStatus() tests                                                  2 ms

   should equal soldout                                       passed     0 ms
   should equal limited                                       failed     2 ms
   should equal available                                     passed     0 ms
```



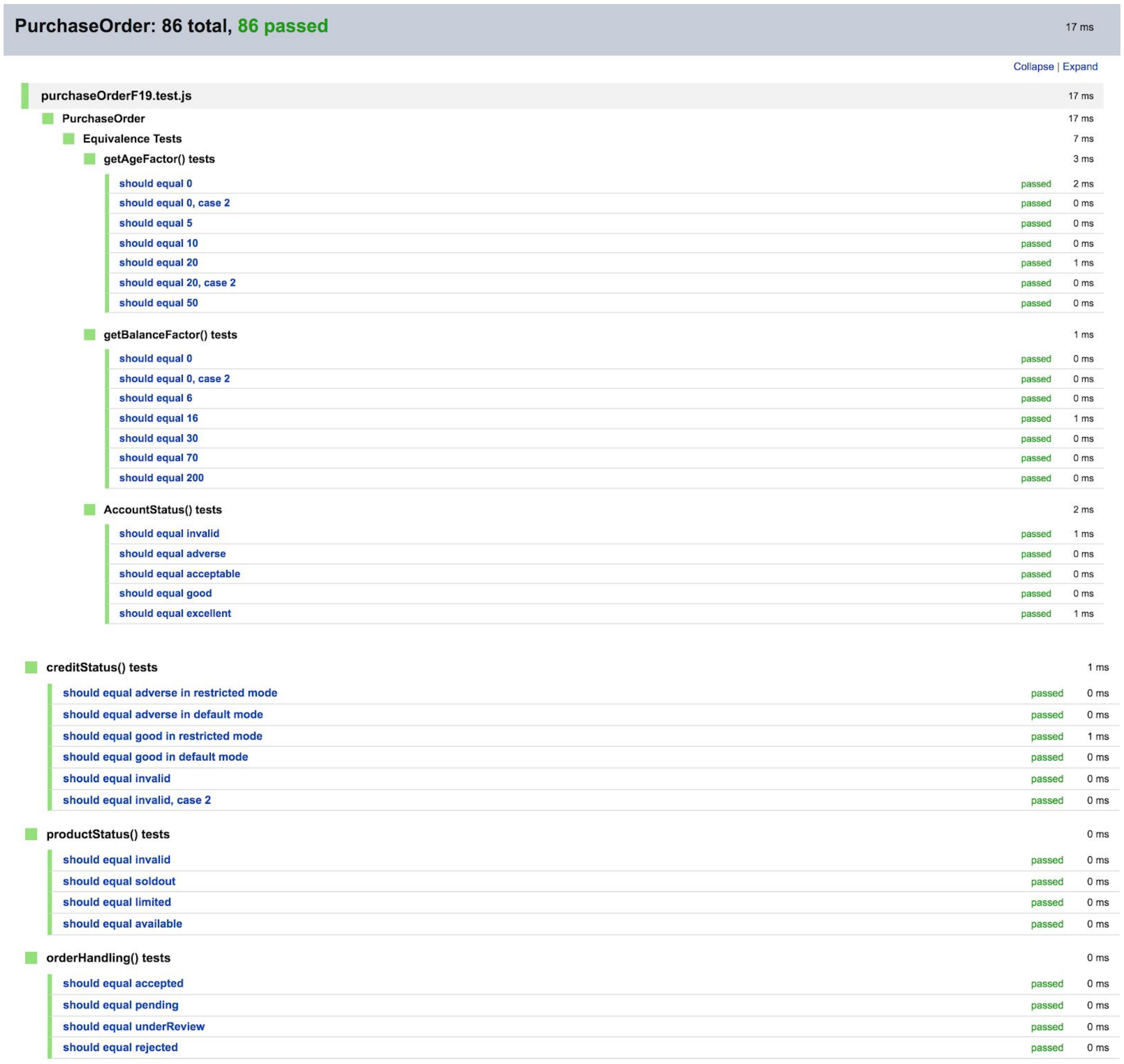
```
   should equal available                                     passed     0 ms
```



```
🟥 Decision Table Testing                                                12 ms
   🟥 orderHandling tests                                                12 ms

      should equal accepted                                   failed     1 ms
      should equal accepted, case 2                           failed     1 ms
      should equal accepted, case 3                           failed     2 ms
      should equal accepted, case 4                           failed     1 ms
      should equal pending                                    failed     1 ms
      should equal pending, case 2                            failed     1 ms
      should equal pending, case 3                            failed     1 ms
      should equal underReview                                failed     2 ms
      should equal underReview, case 2                        failed     1 ms
      should equal rejected                                   failed     0 ms
      should equal rejected, case 2                           passed     1 ms
      should equal rejected, case 3                           passed     0 ms
      should equal rejected, case 4                           passed     0 ms
      should equal rejected, case 5                           passed     0 ms
```

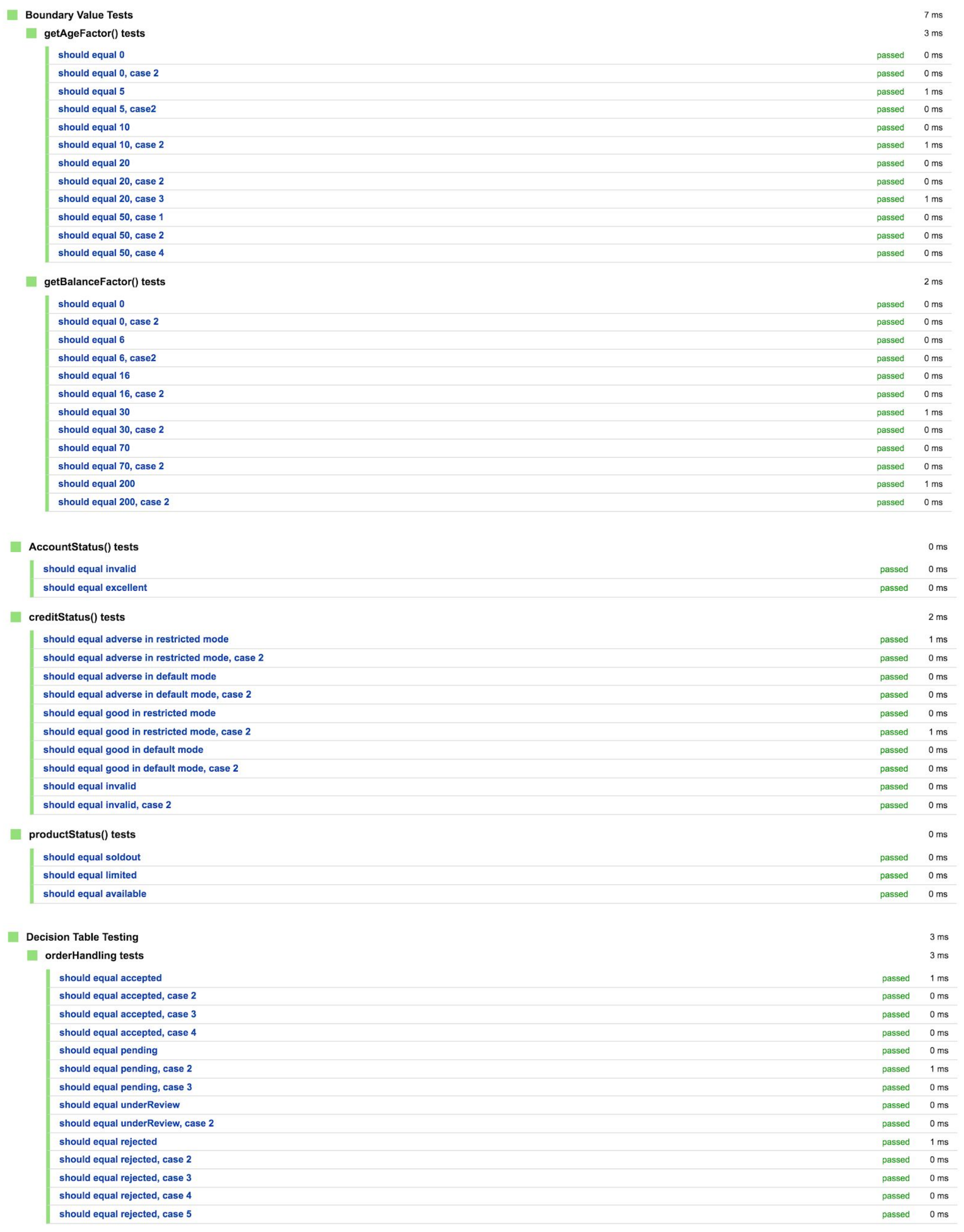
Tomiwa Ademidun

After Code Fixes



```
PurchaseOrder: 86 total, 86 passed                                   17 ms

                                                          Collapse | Expand

purchaseOrderF19.test.js                                             17 ms
  PurchaseOrder                                                      17 ms
    Equivalence Tests                                                 7 ms
      getAgeFactor() tests                                            3 ms
        should equal 0                                       passed   2 ms
        should equal 0, case 2                               passed   0 ms
        should equal 5                                       passed   0 ms
        should equal 10                                      passed   0 ms
        should equal 20                                      passed   1 ms
        should equal 20, case 2                              passed   0 ms
        should equal 50                                      passed   0 ms

      getBalanceFactor() tests                                        1 ms
        should equal 0                                       passed   0 ms
        should equal 0, case 2                               passed   0 ms
        should equal 6                                       passed   0 ms
        should equal 16                                      passed   1 ms
        should equal 30                                      passed   0 ms
        should equal 70                                      passed   0 ms
        should equal 200                                     passed   0 ms

      AccountStatus() tests                                           2 ms
        should equal invalid                                 passed   1 ms
        should equal adverse                                 passed   0 ms
        should equal acceptable                              passed   0 ms
        should equal good                                    passed   0 ms
        should equal excellent                               passed   1 ms

creditStatus() tests                                                  1 ms
  should equal adverse in restricted mode                    passed   0 ms
  should equal adverse in default mode                       passed   0 ms
  should equal good in restricted mode                       passed   1 ms
  should equal good in default mode                          passed   0 ms
  should equal invalid                                       passed   0 ms
  should equal invalid, case 2                               passed   0 ms

productStatus() tests                                                 0 ms
  should equal invalid                                       passed   0 ms
  should equal soldout                                       passed   0 ms
  should equal limited                                       passed   0 ms
  should equal available                                     passed   0 ms

orderHandling() tests                                                 0 ms
  should equal accepted                                      passed   0 ms
  should equal pending                                       passed   0 ms
  should equal underReview                                   passed   0 ms
  should equal rejected                                      passed   0 ms
```

Tomiwa Ademidun



```
🟩 Boundary Value Tests                                                   7 ms
  🟩 getAgeFactor() tests                                                 3 ms
     should equal 0                                            passed     0 ms
     should equal 0, case 2                                    passed     0 ms
     should equal 5                                            passed     1 ms
     should equal 5, case2                                     passed     0 ms
     should equal 10                                           passed     0 ms
     should equal 10, case 2                                   passed     1 ms
     should equal 20                                           passed     0 ms
     should equal 20, case 2                                   passed     0 ms
     should equal 20, case 3                                   passed     1 ms
     should equal 50, case 1                                   passed     0 ms
     should equal 50, case 2                                   passed     0 ms
     should equal 50, case 4                                   passed     0 ms

  🟩 getBalanceFactor() tests                                             2 ms
     should equal 0                                            passed     0 ms
     should equal 0, case 2                                    passed     0 ms
     should equal 6                                            passed     0 ms
     should equal 6, case2                                     passed     0 ms
     should equal 16                                           passed     0 ms
     should equal 16, case 2                                   passed     0 ms
     should equal 30                                           passed     1 ms
     should equal 30, case 2                                   passed     0 ms
     should equal 70                                           passed     0 ms
     should equal 70, case 2                                   passed     0 ms
     should equal 200                                          passed     1 ms
     should equal 200, case 2                                  passed     0 ms

🟩 AccountStatus() tests                                                  0 ms
   should equal invalid                                        passed     0 ms
   should equal excellent                                      passed     0 ms

🟩 creditStatus() tests                                                   2 ms
   should equal adverse in restricted mode                     passed     1 ms
   should equal adverse in restricted mode, case 2             passed     0 ms
   should equal adverse in default mode                        passed     0 ms
   should equal adverse in default mode, case 2                passed     0 ms
   should equal good in restricted mode                        passed     0 ms
   should equal good in restricted mode, case 2                passed     1 ms
   should equal good in default mode                           passed     0 ms
   should equal good in default mode, case 2                   passed     0 ms
   should equal invalid                                        passed     0 ms
   should equal invalid, case 2                                passed     0 ms

🟩 productStatus() tests                                                  0 ms
   should equal soldout                                        passed     0 ms
   should equal limited                                        passed     0 ms
   should equal available                                      passed     0 ms

🟩 Decision Table Testing                                                 3 ms
  🟩 orderHandling tests                                                  3 ms
     should equal accepted                                     passed     1 ms
     should equal accepted, case 2                             passed     0 ms
     should equal accepted, case 3                             passed     0 ms
     should equal accepted, case 4                             passed     0 ms
     should equal pending                                      passed     0 ms
     should equal pending, case 2                              passed     1 ms
     should equal pending, case 3                              passed     0 ms
     should equal underReview                                  passed     0 ms
     should equal underReview, case 2                          passed     0 ms
     should equal rejected                                     passed     1 ms
     should equal rejected, case 2                             passed     0 ms
     should equal rejected, case 3                             passed     0 ms
     should equal rejected, case 4                             passed     0 ms
     should equal rejected, case 5                             passed     0 ms
```