

PRUEBA TECNITCA QA AUTOMATION

JUAN DAVID CORONADO DUSSÁN

- COMPONENTE PRÁCTICO (Funcional)

Creación de la tabla Students

CREATE TABLE Students (

```
student_id INT,  
name VARCHAR(100),  
birthdate DATE,  
gender CHAR(1),  
email VARCHAR(100),  
phone VARCHAR(20)
```

);

```
INSERT INTO Students (student_id, name, birthdate, gender, email, phone) VALUES  
(9000185, 'PEDRO GOMEZ', '1995-10-30', 'M', 'PEDRO.GOMEZ@GMAIL.COM', '1185432478'),  
(9000186, 'JUAN HERNANDEZ', '1995-01-25', 'M', 'Juan.Hernandez@gmail.com', '1185432479'),  
(9000187, 'MARIA GONZALEZ', '1998-08-01', 'F', 'Maria.Gonzalez@hotmail.com', '1185432480'),  
(9000188, 'PABLO SANCHEZ', '1997-03-09', 'M', 'pablo.sanchez@gmail.com', '1185432481'),  
(9000189, 'ELENA CASTRO', '2000-06-15', 'F', 'Elena.Castro@gmail.com', '1185432482'),  
(9000190, 'MATIAS PEREZ', '1998-09-06', 'M', 'Matias.Perez@gmail.com', '1185432483'),  
(9000191, 'SEBASTIAN VARGAS', '2002-08-25', 'M', 'Sebastian.Vargas@outlook.com',  
'1185432484'),  
(9000192, 'PEDRO HERNANDEZ', '1998-05-19', 'M', 'Pedro.Hernandez92@gmail.com',  
'1185432485'),  
(9000193, 'CAROLINA LUNA', '1997-07-21', 'F', 'Carolina.Luna@yahoo.com', '1185432486'),  
(9000194, 'MONICA FIGUEROA', '1980-06-18', 'F', 'Monica.Figueroa@hotmail.com', '1185432487');
```

SELECT * FROM Students;

The screenshot shows the Paiza.io MySQL web interface. The top bar includes the Paiza.io logo, navigation links, and language settings. The main area displays the executed SQL queries and their results. The first query creates the 'Students' table with columns: student_id, name, birthdate, gender, email, and phone. The second query inserts 14 rows of student data. The bottom section shows the output of a 'SELECT * FROM Students;' query, displaying a table with 14 rows of student information.

student_id	name	birthdate	gender	email	phone
9000185	PEDRO GOMEZ	1995-10-30	M	PEDRO.GOMEZ@GMAIL.COM	1185432478
9000186	JUAN HERNANDEZ	1995-01-25	M	Juan.Hernandez@gmail.com	1185432479
9000187	MARIA GONZALEZ	1998-08-01	F	Maria.Gonzalez@hotmail.com	1185432480
9000188	PABLO SANCHEZ	1997-03-09	M	pablo.sanchez@gmail.com	1185432481
9000189	ELENA CASTRO	2000-06-15	F	Elena.Castro@gmail.com	1185432482
9000190	MATIAS PEREZ	1998-09-06	M	Matias.Perez@gmail.com	1185432483
9000191	SEBASTIAN VARGAS	2002-08-25	M	Sebastian.Vargas@outlook.com	1185432484
9000192	PEDRO HERNANDEZ	1998-05-19	M	Pedro.Hernandez92@gmail.com	1185432485
9000193	CAROLINA LUNA	1997-07-21	F	Carolina.Luna@yahoo.com	1185432486
9000194	MONICA FIGUEROA	1980-06-18	F	Monica.Figueroa@hotmail.com	1185432487

creación de la tabla Career_Student

CREATE TABLE Career_Student (

```
career_id INT,  
student_id INT,  
start_date DATE,  
estimated_end_date DATE,
```

$$);$$

```
(10029, 9000185, '2008-01-01', '2013-01-01', '2013-01-01', NULL, 68);
```

```
SELECT * FROM Career_Student;
```

crwr_id	student_id	start_date	estimated_end_date	end_date	extended	calification
10025	9000185	2000-01-01	2013-01-01	NULL		
10023	9000186	2010-01-01	2021-01-01	NULL	92	
10026	9000187	2014-01-01	2018-01-01	NULL	58	
10027	9000188	2017-01-01	2020-01-01	NULL	67	
10028	9000189	2016-01-01	2018-01-01	NULL	91	
10029	9000190	2019-01-01	2022-01-01	NULL	89	
10026	9000191	2020-01-01	2023-01-01	NULL	75	
10029	9000192	2021-01-01	2024-01-01	NULL	74	
10029	9000193	2022-01-01	2025-01-01	NULL	91	
10029	9000185	2000-01-01	2013-01-01	NULL	68	

Creación de la tabla Careers

);

```
INSERT INTO Careers (career_id, name, description, subjects, cost) VALUES
```

```
(10025, 'INGENIERIA', 'INGENIERIA', 90, 25000.00),
(10026, 'ADMINISTRACION', 'ADMINISTRACION', 72, 18000.00),
(10027, 'CONTABILIDAD', 'CONTABILIDAD', 60, 15000.00),
(10028, 'DERECHO', 'DERECHO', 85, 23000.00),
(10029, 'INFORMATICA', 'INFORMATICA', 78, 19500.00),
(10030, 'COMUNICACION', 'COMUNICACION', 65, 17800.00);
```

SELECT * FROM Careers;

Salida	Entrada	Comments
career_id	name	description
10025	INGENIERIA	INGENIERIA
10026	ADMINISTRACION	ADMINISTRACION
10027	CONTABILIDAD	CONTABILIDAD
10028	DERECHO	DERECHO
10029	INFORMATICA	INFORMATICA
10030	COMUNICACION	COMUNICACION

1. Redactar las siguientes consultas en SQL

a) Seleccione el total de estudiantes Masculinos Registrados

SELECT COUNT(*) AS total_masculinos FROM Students WHERE gender = 'M';

Salida	Entrada	Comments
total_masculinos		
6		

b) Elimine los estudiantes nacidos antes 2000

DELETE FROM Students WHERE birthdate < '2000-01-01';

```
111
112
113
114 DELETE FROM Students WHERE birthdate < '1990-01-01';
115
116
117
118
119
```

Ejecutar (Ctrl-Enter)

Salida Entrada Comentarios

(1.37 sec)

- c) Actualice la descripción de la carrera con id '10026' de "Administración" a "Administración de Empresas"

UPDATE Careers SET description = 'Administración de Empresas' WHERE career_id = 10026;
SELECT * FROM Careers;

```
132
133
134 UPDATE Careers SET description = 'Administración de Empresas' WHERE career_id = 10026;
135
136
137
138 SELECT * FROM Careers;
139
140
141
142
143
144
145
146
147
148
149
150
```

Ejecutar (Ctrl-Enter)

Salida Entrada Comentarios

(1.37 sec)

career_id	name	description	subjects	cost
10025	INGENIERIA	INGENIERIA	90	25000.00
10026	ADMINISTRACION	Administración de Empresas	72	18000.00
10027	CONTABILIDAD	CONTABILIDAD	60	15000.00
10028	DERECHO DERECHO	DERECHO	85	23000.00
10029	INFORMATICA	INFORMATICA	78	19500.00
10030	COMUNICACION	COMUNICACION	65	17000.00

- d) Muestre los nombres de los estudiantes que estén cursando dos o más materias

SELECT s.name FROM Students s JOIN Career_Student cs ON s.student_id = cs.student_id
GROUP BY s.student_id, s.name HAVING COUNT(cs.career_id) >= 2;

```
152
153
154
155 SELECT s.name FROM Students s JOIN Career_Student cs ON s.student_id = cs.student_id GROUP BY s.student_id, s.name HAVING COUNT(cs.career_id) >= 2;
156
157
158
159
160
161
162
163
164
165
166
167
168
169
```

Ejecutar (Ctrl-Enter)

Salida Entrada Comentarios

(1.37 sec)

name
PEDRO GOMEZ

- e) Calificación promedio de todos los estudiantes que estén cursando carreras

SELECT AVG(calification) AS promedio_calificaciones FROM Career_Student;

```
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
25
```

```
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

Ejecutar (Ctrl+Enter)

Salida Entrada Comments

name	career_id	email
PEDRO GOMEZ	10025	Pedro.Gomez@gmail.com
JUAN HERNANDEZ	10023	Juan.Hernandez@gmail.com
ELENA CASTRO	10028	Elena.Castro@gmail.com
MATIAS PEREZ	10025	Matias.Perez@gmail.com
SEBASTIAN VARGAS	10026	Sebastian.Vargas@outlook.com
PEDRO HERNANDEZ	10029	Pedro.Hernandez92@gmail.com
CAROLINA LUNA	10025	Carolina.Luna@yahoo.com

g) Cuántos estudiantes que estén cursando 2 o más carreras
SELECT COUNT(*) AS estudiantes_multicarrera FROM (SELECT student_id FROM Career_Student GROUP BY student_id HAVING COUNT(career_id) >= 2) AS subquery;

```
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

Ejecutar (Ctrl+Enter)

Salida Entrada Comments

estudiantes_multicarrera
1

2. De la siguiente imagen, reporte al menos un bug que incluya todos los datos que considere relevantes para que un desarrollador pueda replicarlo y solucionarlo

ID del Bug: BUG-PAGO-001

Título: Duplicación del método de pago "RappiCuenta" en la pantalla de selección

Fecha: 2025-10-04

Reportado por: QA Tester

Descripción:

En la pantalla de selección de métodos de pago, el método "RappiCuenta" aparece listado dos veces: una como "RappiCuenta o débito" y otra como "RappiCuenta", ambos con la misma descripción: "¡Pago fácil e inmediato!". Esta duplicación puede generar confusión en el usuario respecto a cuál opción debe seleccionar y sugiere un problema de diseño o lógica de renderizado.

Pasos para reproducir:

1. Abrir la app de Rappi.
2. Iniciar un pago.
3. Proceder a la pantalla de selección de método de pago.
4. Observar las opciones listadas.

Resultado esperado:

Cada método de pago debería aparecer una sola vez, con su nombre claro y sin ambigüedad, evitando duplicados innecesarios o confusos.

Resultado actual:

Se presentan dos opciones visualmente similares:

- "RappiCuenta o débito"
- "RappiCuenta"

Ambas con descripciones idénticas y sin aclarar sus diferencias, si las hay.

Entorno:

- Dispositivo: Android / IOS
- Versión de la app: 1.7.3
- Pantalla: Método de pago

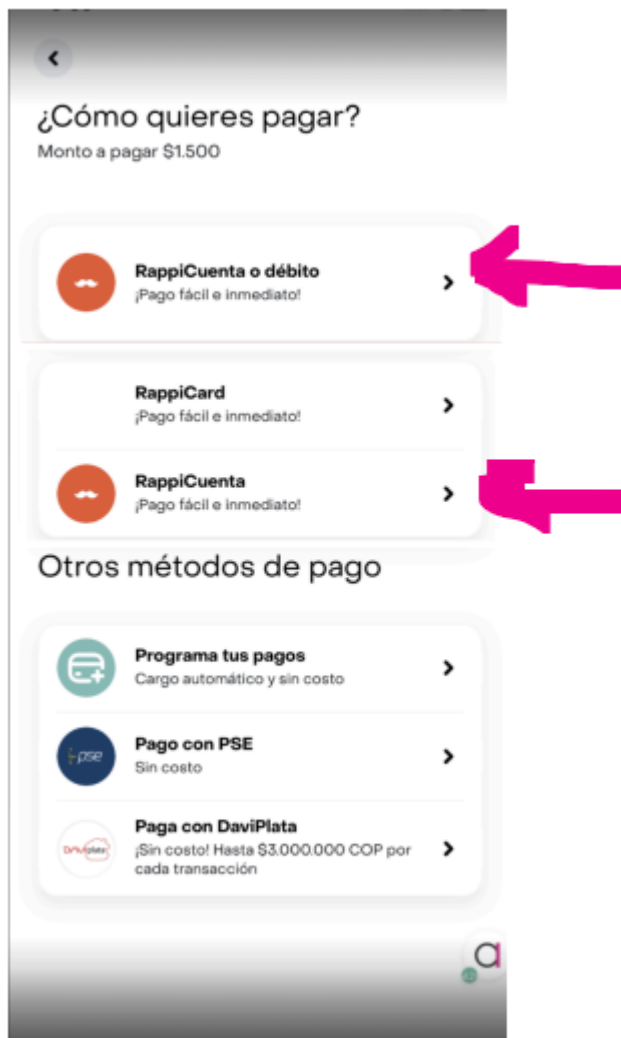
Severidad: Media

- Puede causar confusión en el usuario, pero no bloquea el flujo de pago.

Prioridad: Alta

- Es un tema de experiencia de usuario que puede impactar negativamente la percepción del sistema y generar errores en la selección del método de pago.

Imagen del bugs:



- COMPONENTE TEÓRICO (Automation)

Basado en su experiencia, responda las siguientes preguntas:

1. ¿Cuáles son las principales diferencias entre automatizar en Android vs iOS con Appium?

ASPECTO	IOS	ANDROID
Requisitos del Sistema Operativo	Solo funciona en macOS (por requerimiento de Xcode).	Puede ejecutarse en Windows, macOS o Linux.
Instalación del entorno	Necesitas Xcode, Xcode Command Line Tools, y un perfil de desarrollador.	Necesitas Android SDK, Java JDK, emuladores, y configurar variables de entorno.
Dispositivos y Emuladores	Emuladores más limitados; requieren simuladores Xcode y perfiles de confianza.	Fácil de ejecutar en emuladores sin restricciones.
Interacción con el sistema	Permite más acciones (ej: manipular notificaciones, cambiar red)	Más limitado por seguridad del sistema
Firma de Aplicaciones	Puedes usar APKs sin firmar o firmarlos tú mismo.	Las apps deben estar firmadas con certificados válidos y, para pruebas en dispositivos reales, deben estar en el perfil del equipo de desarrollo.

2. ¿Cómo diseñaría un framework de automatización para mobile multiplataforma que sea mantenible y escalable?

Se propone el uso de Java 11+, Appium como motor de automatización y un framework de pruebas que puede ser:

- Serenity + Screenplay + Cucumber
- TestNG/JUnit

Principios de diseño (SOLID)

El framework aplica los principios SOLID para garantizar mantenibilidad y extensibilidad:

- S – Single Responsibility: cada clase cumple una única responsabilidad (ejemplo: DriverFactory solo crea instancias de drivers, LoginPage solo modela la UI de login).
- – Open/Closed: la arquitectura permite agregar nuevas plataformas o flujos sin modificar el código existente (ej. agregar un IOSCapabilitiesProvider).
- L – Liskov Substitution: las interfaces y abstracciones permiten intercambiar implementaciones (ej. Android/iOS) sin afectar a los consumidores.
- I – Interface Segregation: se definen interfaces pequeñas y específicas, evitando dependencias innecesarias.
- D – Dependency Inversion: las pruebas dependen de abstracciones, no de implementaciones concretas (ej. CapabilitiesProvider en lugar de DesiredCapabilities directo).

Arquitectura propuesta

Capas principales

1. Tests
 - Escenarios en JUnit/TestNG o Features en Cucumber.
 - Simulan los flujos de usuario.
2. Tasks / Interactions (Screenplay)

- Encapsulan acciones (ej. LoginTask).
- En TestNG se puede usar una capa de “Helpers” con métodos reutilizables.
- 3. Pages
 - Modelos de pantallas móviles con localizadores multiplataforma (@AndroidFindBy, @iOSXCUITFindBy).
- 4. Platform / Capabilities
 - Configuración desacoplada por plataforma (Android/iOS).
- 5. Utils
 - Esperas, gestos, generación de datos, logs.
- 6. Config
 - Archivos de propiedades (.properties, .yaml) para entornos locales y remotos.
- 7. Reports
 - Serenity: reportes detallados paso a paso, ideal para Screenplay.
 - Allure: opción ligera si se usa TestNG/JUnit.
 - extent report: opción ligera si se usa TestNG/JUnit.

Escalabilidad y ejecución

- Paralelismo mediante ThreadLocal drivers y ejecución en paralelo con TestNG o Serenity.
- Dependencias de Maven/Gradle.
- CI/CD con GitHub Actions, Jenkins o GitLab CI para ejecución automática.
- Integración con device farms para ampliar cobertura de dispositivos y versiones.

Buenas prácticas

- Centralizar localizadores y configuración.
- Evitar Thread.sleep(), usar esperas explícitas.
- Mantener independencia entre pruebas.
- Definir nomenclatura estándar para clases y escenarios.

3. ¿Qué criterios usa para seleccionar qué casos se deben automatizar y cuáles no?

Criterios para automatizar:

- Alta frecuencia de ejecución.
- Casos críticos para el negocio.
- Repetitivos y estables en el tiempo.
- Funcionalidades ya estabilizadas.
- Costos y beneficios.

Criterios para NO automatizar:

- Flujos que cambian con frecuencia.
- Procesos muy esporádicos o de bajo volumen.
- Casos que requieren interacción física (ej: huella dactilar).
- Pruebas exploratorias o de usabilidad.

4. ¿Qué estrategias aplicaría para integrar pruebas automáticas en un pipeline CI/CD?

- Cual herramienta usar para la creación del pipeline
- Diseña un pipeline con etapas claras (Build, ejecución de prueba, sonar, escaneo de vulnerabilidades)

- Ejecución por rama
- integración con granjas de dispositivos remotas
- Paralelización y optimización del tiempo de ejecución
- Resultados automáticos
- Notificaciones
- Integrar con los desarrollos

5. Mencione las ventajas y desventajas de trabajar con dispositivos físicos vs. dispositivos en remoto / granja de dispositivos cuando ejecuta una suite de automatización.

Factor	Dispositivos Físicos	Granja de Dispositivos (Remoto)
Costo	Alto (compra, mantenimiento)	Pago por uso, escalable
Acceso y control	Total (red, logs, debugging, gestos)	Limitado a lo que permite el proveedor
Paralelismo	Limitado por la cantidad de dispositivos físicos	Alta escalabilidad
Configuración inicial	Compleja pero controlada	Configuración ya proporcionada
Integración CI/CD	Requiere hardware conectado al servidor	Integración directa con servicios en la nube

6. ¿Qué patrones de diseño ha implementado en sus proyectos que considere eficaces para minimizar la duplicación de código y mejorar la legibilidad al desarrollar automatizaciones para dispositivos móviles?

- Page Object Model (POM)
- Serenity Screenplay + Cucumber
- COMPONENTE PRÁCTICO (Automation)

Contexto: Se requiere implementar un nuevo script para una prueba automatizada. El caso de prueba involucra tres (3) vistas: Home del App, Login y el Home de producto. El flujo que se desea automatizar valida que el usuario Ingrese al Home del App donde da un tap al widget del producto para posteriormente ser redireccionado al Login. Allí ingresará un código de 6 dígitos y finalmente navegará de forma automática al Home del producto donde se encuentra el detalle y acciones que el usuario puede realizar sobre el producto.

- a) En la vista de Login se implementa un método para validar que todos los labels definidos en el diseño se encuentren presentes. A continuación, verá el código implementado.

```
1  public boolean verifyLabelsOnScreen() {
2      try {
3          // Espera para que la pantalla cargue los elementos
4          Thread.sleep(15000);
5
6          boolean flagLabelIniciarSesion = false;
7          boolean flagLabelHola = false;
8          boolean flagLabelIngresarContrasena = false;
9          boolean flagLabelOlvideContrasena = false;
10
11         // Recorre todos los labels en la pantalla
12         for (MobileElement label : screenLabels) {
13             String text = label.getAttribute("text");
14
15             if (text.equals("Iniciar Sesión")) {
16                 flagLabelIniciarSesion = true;
17             }
18             if (text.contains("¡Hola")) {
19                 flagLabelHola = true;
20             }
21             if (text.equals("Ingresa tu contraseña de 6 dígitos")) {
22                 flagLabelIngresarContrasena = true;
23             }
24             if (text.equals("Olvidé mi contraseña")) {
25                 flagLabelOlvideContrasena = true;
26             }
27         }
28
29         // Verifica que todos los labels estén presentes
30         if (flagLabelIniciarSesion && flagLabelHola
31             && flagLabelIngresarContrasena && flagLabelOlvideContrasena) {
32             return true;
33         }
34
35     } catch (Exception e) {
36         return false;
37     }
38     return false;
39 }
40
```

Instrucción: De acuerdo con su experiencia indique si está de acuerdo con la forma en la que fue implementado el método. En caso de no estar de acuerdo. Indíquenos qué modificaciones haría o reescriba el método para ver reflejados sus cambios y justifique su respuesta.

De acuerdo con mi experiencia, con la forma de implementación del código presentado tiene varias mejoras por realizar:

- `EL Thread.sleep(15000)`: es una mala practica que este siempre espere 15 segundos, ya que hace que la prueba se mas lenta, adicional a esto aun que los elementos ya estén cargados y se puedan interactuar con estos antes de 15 segundos este obliga a esperar los 15 segundos
Para ello una buena practica seria usar los `wait` + condición ya solo espera a que el o los elementos estén cargados e interactúa inmediatamente con ellos
 - El uso de las banderas booleanas repetitiva: código repetitivo y agrega complejidad, adicional en temas de escalabilidad y mantenibilidad en el tiempo se vuelve insostenible. Mejor usar una lista de textos esperados y se comparada con los textos encontrados.
 - Uso de `equals`: el riesgo que genera esto es si un label tiene un espacio de más, mayúsculas/minúsculas distintas o un carácter especial, el test puede fallar. Se puede usar `contains` si el label varia o `equalsIgnoreCase` si no mitigando el riesgo.
 - Código repetitivo y uso excesivo del `if` generando complejidad, mantenibilidad y escalabilidad.
Para ello es mejor usar `Set` o `Map` con textos esperados y validarlos en un bucle.
 - Uso de `return false` redundante el último `return false` (fuera del `catch`) es innecesario porque el flujo ya está cubierto.
 - Falta mensajes de error si la validación falla, no sabes qué label faltó. Solo devuelve `false`, para ellos lo ideal es logs o lanzar excepción con el detalle.
 - Falta de reutilización lo cual hace que no sea posible usar en otras partes del código si es requerido.
 - Problema de escalabilidad ya que por ahora son 4 pero si son 20 o 50 hace que el método se vuelve enorme y difícil de leer.
- b) En la vista del Home del App, se necesita crear dos métodos: El primero estará encargado de verificar si el usuario se encuentra o no en el Home del App y el segundo realizará la acción de dar tap en el botón de Rappipay para ingresar al login. A continuación, verá la definición de los elementos que se tuvieron en cuenta.

```

1  @AndroidFindBy(id = "com.grability.rappi:id/menu_image")
2  private MobileElement buttonMenu;
3  @AndroidFindBy(id = "com.grability.rappi:id/support_image")
4  private MobileElement buttonSupport;
5  @AndroidFindBy(className = "android.widget.TextView")
6  private List<MobileElement> widgetElements;
7  @AndroidFindBy(id = "com.grability.rappi:id/imageView_action")
8  private MobileElement buttonRappiPay;
9  @AndroidFindBy(id = "layout_primary_action")
10 private MobileElement BTN_RAPPIPAY_CO;
11 @AndroidFindBy(xpath = "//*[contains(@text, 'Restaurante')]")
12 private MobileElement RESTAURANT;
13 @AndroidFindBy(xpath = "//*[contains(@text, 'Mercado/Ex')]")
14 private MobileElement MARKET;
15 @AndroidFindBy(xpath = "//*[contains(@text, 'Turbo')]")
16 private MobileElement TURBO;
17

```

Instrucción: De acuerdo con su experiencia indique si está de acuerdo con la forma en la que se definieron los elementos. En caso de no estar de acuerdo. Indíquenos qué modificaciones haría o reescriba el código para ver reflejados sus cambios y justifique su respuesta.

De acuerdo con mi experiencia, con la forma de implementación funciona, sin embargo, tiene varias mejor que se pueden realizar:

- Uso de className = "android.widget.TextView" para widgetElements: muy genérico. Capturará todos los TextView de la pantalla haciendo que guarde información no requerida.
- Xpath con // * hace que la automatización se vuelva lenta ya que va a buscar en todo el DOM del código es mejor especificar ejemplo: //a o //h1
- Los xpath con texto son frágiles si cambia el idioma de la app (ES → EN)
- Nombre de las variables no tienen estructura, mezclas mayúsculas/abreviaturas para ello usar naming, ejemplo btnMenu.
- Los nombres de algunas variables no tienen contexto para que va a ser su uso ejemplo: private List<MobileElement> widgetElements;
- Localizadores muy específicos (com.grability.rappi:id/menu_image) hacen que el Page Object sea poco portable si la app cambia la estructura.
- Si estamos usando localizadores como el Id no es una buena práctica com.grability.rappi:id/menu_image es mejor colocar menu_image