Testing Input Validation:

Testing Mail Address Input

Testing validation of mail input with pairwise testing using pict.

Test ID	Mail Address Name >= 3 chars	Name in Legal Characters	Has "@" separator	Mail Domain >= 3 chars	Domain in Legal Characters	Has "." separator	Top Level Domain (TLD) >= 2 chars	TLD within Legal Characters	TLD <= 63 characters
T01	F	T	F	T	F	F	T	F	F
T02	T	F	T	F	T	T	F	T	T
T03	T	T	T	T	T	T	T	F	T
T04	F	F	F	F	F	T	F	T	F
T05	T	T	F	F	F	F	T	T	T
T06	F	F	T	T	T	F	F	F	T
T07	T	T	T	F	T	T	F	F	F
T08	F	F	F	T	T	F	T	T	T
T09	F	F	T	F	F	F	T	F	F

Test Implementation:

https://github.com/SGTK06/Job-Application-Tracker/blob/f39cf96de8ea966d38b4f4f4a5d0883a3db42b8e/tests/test_input_mail_validation.py

Testing Username Input:

Testing username input using both Equivalence Partitioning and Boundary Value Analysis

Tests cases for Equivalence Partition Testing of username input validation logic.
Condition: username must be longer than 5 characters (boundary = 5)
Partitions:
1. Less than mininum number of character
2. More than mininum number of character

Test ID	Username Length	Username Validation (Expected)	Username Validation (Actual)	Pass/Fail
T01	2 (<min length)	FALSE	FALSE	Pass
T02	10 (>min length)	TRUE	TRUE	Pass

Tests cases for Boundary Value Analysis of username input validation logic. Condition: username must be longer than 5 characters (boundary = 5) [….4, 5, 6…..]

Boundary Value = 5, 3-way BVA => test before boundary, test on boundary, test after boundary

Test before boundary: at 4

Test on boundary: at 5

Test after boundary: at 6

Test ID	Username Length	Username Validation (Expected)	Username Validation (Actual)	Pass/Fail
T03	4	FALSE	FALSE	Pass
T04	5	FALSE	FALSE	Pass
T05	6	TRUE	TRUE	Pass

Test Implementation:

https://github.com/SGTK06/Job-Application-Tracker/blob/f39cf96de8ea966d38b4f4f4a5d0883a3db42b8e/tests/test_input_username_validation.py

Testing Skills Input:

Testing skills input using both Equivalence Partitioning and Boundary Value Analysis

Tests cases for Equivalence Partition Testing of skills input validation logic. Condition: number of skills must be more than 5 (boundary = 5)

Partitions:

1. Less than mininum number of skills
2. More than mininum number of skills

Test ID	Number of Skills	Skills Validation (Expected)	Skills Validation (Actual)	Pass/Fail
T01	2	FALSE	FALSE	Pass
T02	8	TRUE	TRUE	Pass

Tests cases for Boundary Value Analysis of skills input validation logic. Condition: number of skills must be more than 5 (boundary = 5) [….4, 5, 6…..]

Boundary Value = 5, 3-way BVA => test before boundary, test on boundary, test after boundary

Test before boundary: at 4

Test on boundary: at 5

Test after boundary: at 6

Test ID	Number of Skills	Skills Validation (Expected)	Skills Validation (Actual)	Pass/Fail
T03	4	FALSE	FALSE	Pass
T04	5	TRUE	TRUE	Pass
T05	6	TRUE	TRUE	Pass

Edge Cases:

Test ID	Number of Skills	Skills Validation (Expected)	Skills Validation (Actual)	Pass/Fail
T06	blank whitespace input ' '	FALSE	FALSE	Pass
T07	repeating same skills (6 but 4 unique)	FALSE	FALSE	Pass

Test Implementation:

https://github.com/SGTK06/Job-Application-Tracker/blob/f39cf96de8ea966d38b4f4f4a5d0883a3db42b8e/tests/test_input_skills_validation.py