		Acceptance Criteria:								
		<ul> <li>AC1 - The service MUST process collections of strings i.e. it can proce</li> </ul>	AC1 - The service MUST process collections of strings i.e. it can process 1 string or many string, and return a collection of processed strings							
		AC2 - The service will accept strings that are any length								
		<ul> <li>AC3 - The service will accept strings that can contain any characters</li> </ul>								
		<ul> <li>AC4 - The service MUST not output NULL or empty strings</li> </ul>								
		<ul> <li>AC5 - The service MUST truncate strings that are greater than 15 cha</li> </ul>	AC5 - The service MUST truncate strings that are greater than 15 characters i.e. it MUST process strings and return/output strings that are 15 characters in length or less							
		AC6 - The service MUST replace contiguous dupicate characters with	ACG - The service MUST replace contiguous dupicate characters with a single character - this refers to characters that of the same case sensitivity ONLY i.e. W and W are treated as different characters, therefore are not duplicates of each other							
Project Name:	LDC	AC7 - The service MUST replace any dollar signs with pound signs								
Service Name:	String Collection Processing Service	AC8 - The service MUST remove any underscores								
Created by:	Andy Short	AC9 - The service MUST remove any number 4 characters								
Date of creation:	15/10/2023									
		University Providence and			CT	ATLIC				

TEST CASE ID	TEST SCENARIO	TEST CASE	Linked to Requirement/ Acceptance Criteria	TEST STEPS	TEST DATA	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)		
TC0001	Verify that the service can correctly process a collection containing a single string	Send a collection containing a single string	AC1	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  { "Ac9123KL@%cWw7" }	Output: { "Ac9123KL@%cWw7" }				
TC0002	Verify that the service can correctly process a collection containing more than one strings	Send a collection containing two string	AC1	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'Strings Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:- { "Ac9123KL@%cWw7", "rfcKL@%cWw7", }	Output: {     "Ac9123KL@%cWw7",     "rTeKL@%cWw7", }				
TC0003	Verify that the service can correctly process a collection containing more than 9 strings	Send a collection containing 10 strings	AC1/AC3	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:  {	Output:  {     "Ac9123KL@%cWw7",     "frekL@%cWw7",     "frekL@%cww7",     "Ac9123KL@%crs7",     "Ac9123KL@%crs7",     "Ac9123KL@%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%",     "Ac9123KL@%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%c%",     "Ac9123KL@%c%c%",     "Ac9123KL@%c%c%",     "				
TC0004	Verify that the service can process strings that are more than 15 characters in length, and returns these strings as truncated to 15 characters in length	Send a collection containing strings that are greater than 15 characters in length	AC2	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  { "Ac9123KL@%cWw7P6E(", "rTeKL@cfutf£irsIW7" }	Output:  { "Ac9123KL@%cWw7P", "rTeKL@c{utfEJrs" }				
TC0005	Verify that the service can process strings that are of a minimum length i.e. 1 character long	Send a collection containing strings that are 1 and 2 characters in length	AC2	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  {     "A",     "b",     "1",     "2",     "2" }	Output:  { "A", "b", "a", ""eE", ""zz" }				
TC0006	Verify that the service removes contiguous duplicate characters	Send a collection containing strings with contiguous duplicate characters	AC6	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	input Collection to send to Service: { "AAcC99123KL@%cWWw7P6E(", "TheekL@@cC(utfEirs!W7" }	Output: { "AcC9123KL@%cWw7", "rTteKL@cC{utf£J" }				
TC0007	Verify that the service removes dollar sign characters and replaces them with pound sign characters	Send a collection containing strings with dollar signs	AC7	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection to send to Service:- {     "Ac\$123KL\$\$cW\$£(253",     "\$\$\$55\$\$",     "\$\$55\$\$", }	Output: { "Ac£123KL£cWE(25", "£", }				
TC0008	Verify that the service removes underscores characters	Send a collection containing strings with underscores	AC8	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection to send to Service:  { "Ac_\$12_3KL_cW\$£", "A-b.", "" }	Output: { "Ac\$123KLcW\$£", "A-b" }				

TC0009	Verify that the service removes any number '4' characters	Send a collection containing	AC9	1. Using Postman, send the payload object	Input Collection to send to Service:-	Output:		
		strings containing the number 141		containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	{     "A4c9441243KL4@4%cWw7",     "a643214",     "444444"     } }	{ "Ac9123KL@%cWw7", "aB321" }		
TC0010	Verify that the service processes empty string values correctly	Send a collection of 5 strings, with the string at index position 3 in the collection being an empty string	AC4	Using Postman, send the payload object containing the collection of strings in the ITST DATA column cell to the String Collection Processing 'service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  {	Output: Contains 4 strings:- {   "Ac9123KL@%cWw7",   "a89123KL@%cWw9",   "Gh9123KL&%cWw3",   "OStHfkt[]%cWw1", }		
TC0011	Verify that the service processes NULL values correctly	Send a collection of 5 strings, with the string at index position 3 in the collection being is a NULL value	AC4	1. Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the String Collection Processing 'service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  {  "Ac9123KL@%cWw9",  "B99123KL@%cWw9",  NULL,  "Gh9123KL8 "%cWw3",  "Ok5HHKL()%cWw1",  }	Output: Contains 4 strings:- {     "Ac9123KL@%cWw7",     "a69123KL@%cWw9",     "Gh9123KL&%cWw3",     "OSSHfKL[)%cWw1", }		
TC0012	[EDGE CASE] Verify that the service can gracefully deal with sending strings containing double quotation marks	Send a collection of strings that contain double quotation marks	N/A	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	input Collection of strings to send to Service:-  {  "Ac9"12"3KL@"%cWw7",  """"  "/"""  "/"""  }	Output: {     "Ac9"12"3KL@"%cW",     """"     """"     """"     """" }		
TC0013	[EDGE CASE] Verify that the service can gracefully deal with sending an empty collection	Send an empty collection#	N/A	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:- { }	Output: { } Possible payload would contain a message saying empty payload was sent		
TC0014	[EDGE CASE] Verify that the service can gracefully deal with sending strings containing curly brackets	Send a collection of strings that contain double quotation marks	N/A	Using Postman, send the payload object containing the collection of strings in the TEST DAIA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  {     "("12"3KL@"%cw)",     "(1)",     "(3)",     "(4)", }	Output: {     "("12"3KL@"%cw}",     "(1)",     "(1)",     "(42)3)" }		
TC0015	[EDGE CASE] Verify that the service can gracefully deal with a request containing NO collection payload object	Send a request to the service with no collection of strings payload	N/A	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	No payload	Output:  Service MUST return a 40x error with a message stating request payload was missing		
TC0016	[EDGE CASE] Verify that the service can gracefully deal with a request containing missing commas and quotation marks	Send a malformed collection of strings payload to the service	N/A	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EMPECTED RESULT column cell	Input Collection of strings to send to Service:- { 123tgf*&^, "986rfg" "123, "," }	Output: Service MUST return a 40x error with a message stating request payload is malformed		
TC0017	[EDGE CASE] Verify that the service can handle stings with white spaces in them	Send a collection of strings that containwhite spaces within them	N/A	Using Postman, send the payload object containing the collection of strings in the TEST DATA column cell to the 'String Collection Processing' service endpoint 2. Verify that the returned response body contains the collection of strings in the EXPECTED RESULT column cell	Input Collection of strings to send to Service:-  {  "Ac9123KL@%cW", "aB912 SKL@%", "Gh912XKL*%w3", "Ok5tHHKL()%", }	Output: {     "A_9123KL @%cW",     "aB912 3KL@%",     "Gh912KKE *%w3",     "Ok5tHfKL[]% ", }		