

Transforming test cases to action driven format using NLP

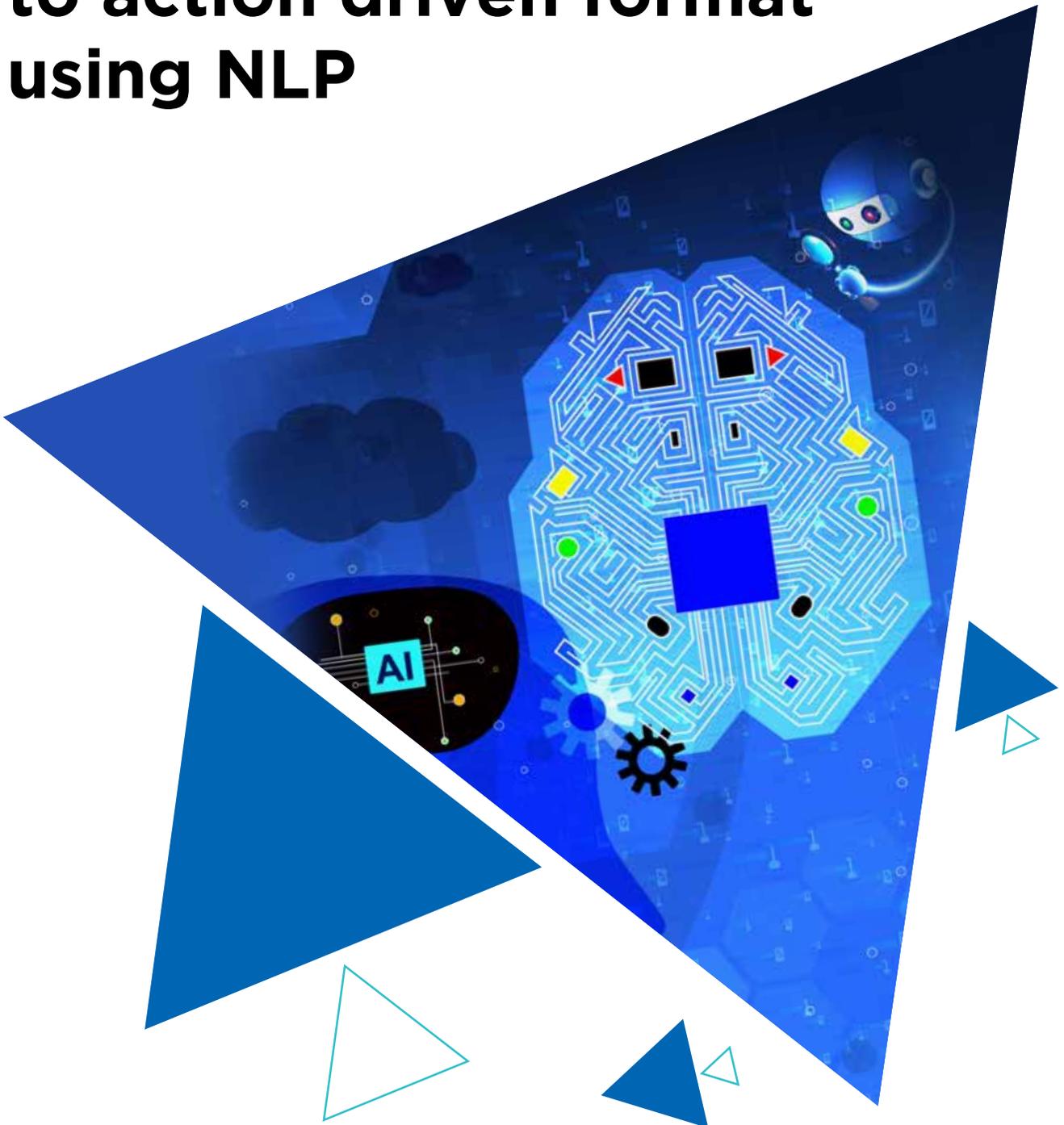


TABLE OF CONTENTS

Abstract	3
Introduction	4
Market Trends	5
Solution	6
Benefits	9
Conclusion	9
Reference	10
Author Info	11

Abstract

Software Testing Life Cycle (STLC) is a testing process which is executed in a sequence, to meet the software quality goals. In today's world with the advent of technologies, companies follow various testing methods to cater to their customer needs. The traditional methods involve manual effort to develop an automation test script from a test case document which involves cost and time. The test script can be executed through Automation frameworks.

With the boom in Artificial Intelligence and Deep learning techniques, the idea is to leverage and adapt NLP techniques in the software testing cycle. This white paper proposes an NLP based tool to mine and process the required information from the test case document. The information is then populated in an Action driven template as an outcome. This will act as an input for enabling execution with Automation Framework and will replace the need for test scripts. The proposed NLP tool helps to overcome the efforts spent in developing test scripts from test case document for automation.

The proposed tool is also known as TAROT (TrAnsfoRmation Of Test case documents) Engine, is designed in a manner that can be plugged into Test Automation frameworks with customization for quick turnaround. This, in turn, helps to accelerate the automation cycle. The TAROT Engine is built with the learning capability to improve automation efficiency from its previous experience.

Introduction

In Software Testing Life Cycle (STLC) [1], the test case development phase is a crucial part. The objective of this phase is to create, review, and rework the test cases to deliver the test scripts on time. The entire testing phase contains diverse activities, techniques, and complex challenges. We see that in the present world, the software is growing continuously to meet the critical, prevalent and complex customer requirements. In parallel, the quality of the software is to be sustained without compromising the test case development efforts. This becomes progressively difficult and expensive.

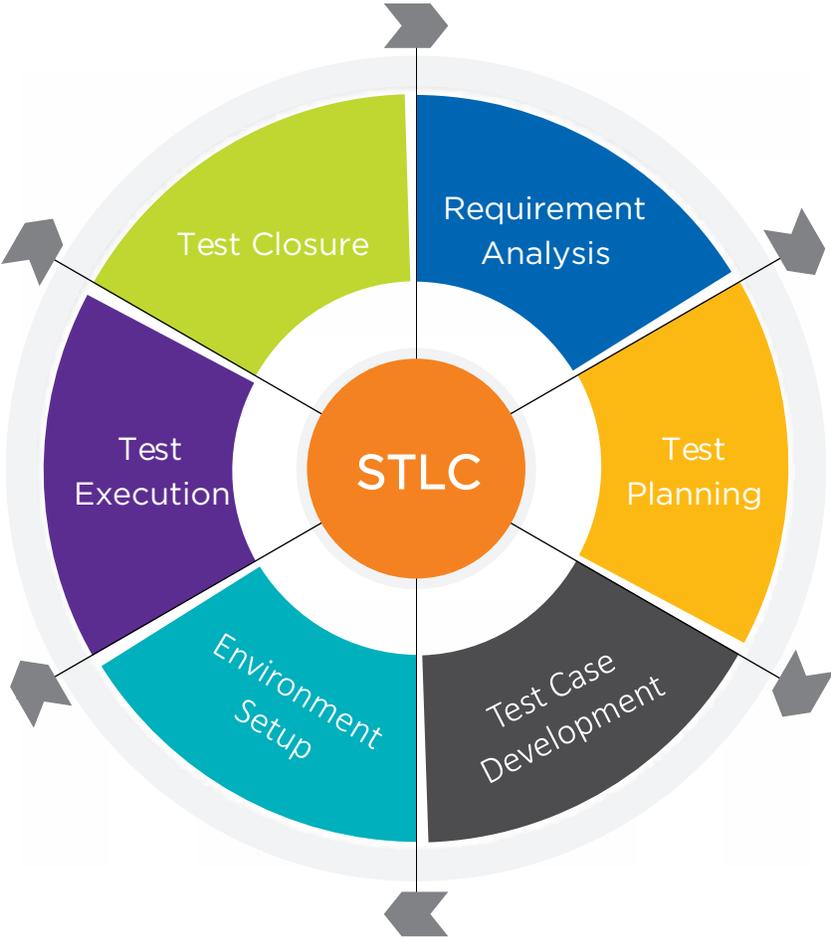


Figure 1- STLC Process

The testing phase can be divided into two categories manual and automated testing. The outcome of transforming manual test steps into automation is a test script. Automation testing makes use of software tools to execute these test scripts. In current trends, Software testing teams are restricted by deadlines and limited resources for testing. To transcend this challenge, most companies are transforming towards AI-based solutions in automated testing.

There is a surplus of Automation Frameworks [2] available in the market, which will assist in creating test scripts, its execution, and analyze the test results. The Automation team develops scripts using libraries, APIs, or subroutines, which are made available in the framework. Most frameworks require effort from the team to get this scripting task completed.

NLP (Natural Language Processing) [3] methods, which are used to process and convert test cases on a large scale without human intervention will be indispensable for understanding these test cases. This approach helps in identifying the words in a sentence (Test case document) to predefined entities such as Control, Control Text, Test Data, and Event. The predefined Action template (Table 2), which acts as an output, will be filled with the entities derived from the Test case document. This is used as an input for an automation framework. With this approach, we can minimize the scripting efforts of the Automation test engineer significantly. The solution improves efficiency and reduces overall turnaround time in the scripting phase.

The TAROT Engine is designed such that it can work with keyword or data-driven Automation Frameworks. The Engine may need minimal customization to transform test cases to the required Action template for test execution.

Market Trends

The software testing community has experienced many trends taking shape in artificial intelligence (AI) and automation.

The current trends include the adoption of powerful test automation practices and tools to meet the increasing test automation demands for web applications. The need for more test automation tools is to manage difficulties in test data and environments, shortening release schedules, and lack of time for testing [4].

Given the recent advancements in applications of Artificial Intelligence and Machine Learning algorithms, there will be scope for more intelligent Test automation technologies and tools. These approaches are being used in various processes of the software testing mechanisms like test sequence generation, automation of testing, and checking the reliability of the system. They would also bring advancements to formulating test scenarios, predicting application behaviors, and predicting areas and levels of test [5].

Solution

As a part of the test case development phase in any project, test case documents will be developed into automation test scripts. There is considerable time and effort spent by the automation scripting team to get the scripts developed. The solution conveys how test case documents are transformed into Action driven template (Table 2) without developing scripts. Thus, a test case document plays a key role in this solution.

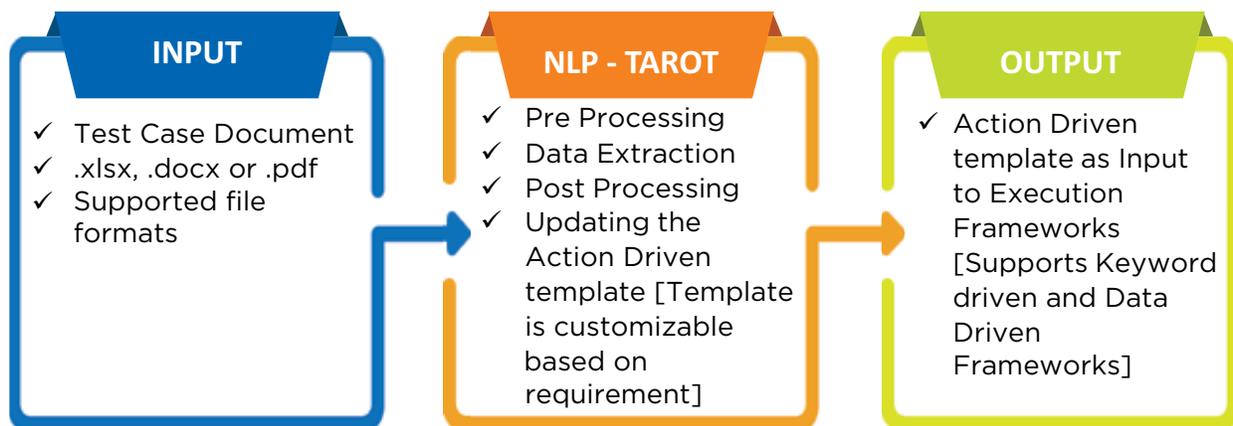


Figure 2: Solution Workflow

Test case documents (Table 1) layout will be available based on project guidelines. As a part of solution development, various formats of test case documents were collected for better understanding and to manage the deviations. The below-mentioned factors were observed.

1. Projects may use any of the tools like TFS, HP ALM, test link, etc. where test cases will be available in a standard template with an option for download.
2. In most cases, there is no common template followed for test case documents. The test case documents will be in any of the file formats like .docx, .xlsx or .pdf.
3. Most of the projects follow a single test case per document and in a few cases, multiple test cases are written in a single document.

The solution is designed to support the above cases effectively.

The TAROT Engine is developed using open-source NLP platforms and packages. The TAROT Engine can parse the test case details and extract the required information. To achieve this, the solution is developed using prevalent Machine Learning and NLP Algorithms [6,7,8].

The Engine expects the test case details in granular nature for optimum performance. At times, testers may even write obscure test steps which may be complex for processing and may need a user review for any corrections. In such cases, it is recommended to deconstruct the test steps into simpler ones for better execution.

We have used a different type of test case documents during development and proof of concept of the solution. The test cases mostly dealt with are web services for News, Telecom, E-commerce, and ERP based applications.

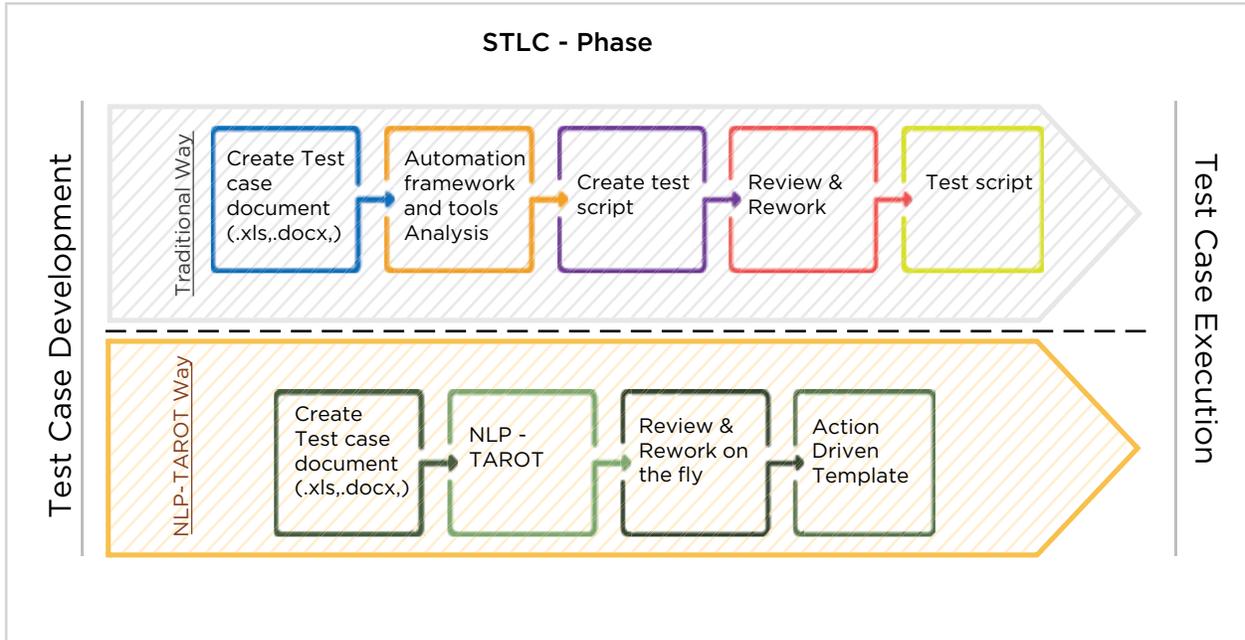


Figure 3: Traditional vs TAROT way

Example:

I) Test case document

Sl.No	Test Step	Test Data
1	Enter the URL http://www.abcdefg.com	
2	Select "Unavailable" radio button	
3	Enter password in "Password" textbox	Password:****
4	Click on "Price" link	
5	Enter price in "Price" textbox	Price:20.00
6	Select currency from "Currency" dropdown	Currency:GBP
7	Enter Tax details in "TAX Details" textbox	Tax Details:NA
8	Enter details in "Expires After" textbox	Expires After:60
9	Click on "Display Address & Contact information on price page" checkbox	
10	Enter time in "Minutes" textbox	Minutes:30
11	Click on "Allow save and resume later" checkbox	
12	Select "Select Theme" dropdown box	Select Theme:Pre Employment Theme

Table 1

II) TAROT - Action Driven Output Template

Control	Control text	Event	Data
toUrl		navigate	http://www.abcdefg.com
Radio	Unavailable	select	
Textbox	Password	enterText	****
Link	Price	click	
Textbox	Price	enterText	20
Dropdown	Currency	select	GBP
Textbox	TAX Details	enterText	NA
Textbox	Expires After	enterText	60
Checkbox	Display Address & Contact information price page	click	
Textbox	Minutes	enterText	30
Checkbox	Allow save and resume later	click	
Dropdown	Select Theme	select	Pre Employment Theme

Table 2

Test case varies with test data given in each input sheet. Based on the content, the solution groups the data to relevant Action items. This is achieved by in-house NLP algorithms. The solution captures the Action items from model and populates to a pre-defined Action template. In case the solution comes across the “Repeat” word as part of the test step, the solution takes appropriate steps without impacting the test coverage. This is used to “Repeat” a set of input statements where the data differs with every iteration (Data provided through external excel files).

There is a possibility to get some of the entities in the Action template to be empty. In such scenarios, the solution uses relevance and correlation techniques to fill in the empty entities in the predefined template. As a part of standardization, the solution validates the entities in the template, to meet the framework requirements. This is because each framework will have its libraries to perform functions. As part of customization, new APIs can be developed based on requirements and can be incorporated into the Framework. The populated template with entities will be shared to the user for final review before test execution by the framework.

Benefits

1. Scripting efforts will be made trivial
2. 3x speed in transforming test case document to a template
3. Learning through previous experience to achieve improvement in performance
4. Tester does not require any coding expertise
5. Improved turnaround time for test case transformation
6. Plug and play capability with automation frameworks
7. Provision to support External Test Data documents.
8. Capability to Process Test document of different file formats (E.g. .docx, .xlsx, pdf, etc.)
9. Inbuilt capability to handle the 'Repeat' statement

Conclusion

This solution converts the Test case document into a pre-defined template using NLP. This is a separate tool that can be plugged into any Automation Framework with minimum integration effort. With this solution, the test case documents can be processed and directly executed without developing test automation scripts, thus, minimizing the scripting efforts and reducing the turnaround time in the automation cycle. As a part of an ongoing enhancement to the solution, Deep Learning techniques will play a vital role in pre and post-processing stages of the solution.

Reference

1. Software Testing Life Cycle <https://www.softwaretestingmaterial.com/stlc-software-testing-life-cycle/>
2. Test Automation Framework <https://www.cs.colorado.edu/~kena/classes/5828/s12/presentation-materials/ghanakotagayatri.pdf>
3. Natural Language Processing https://en.wikipedia.org/wiki/Natural_language_processing
4. Software Testing Trends for 2018 and Beyond <https://dzone.com/articles/software-testing-trends-for-2018-and-beyond>
5. Software Testing Trends of 2018 | The Automation Perspective <http://www.softwaretestingstudio.com/14-software-testing-trends-2018/>
6. A Survey on Recent Advances in Named Entity Recognition from Deep Learning models <https://aclweb.org/anthology/C18-1182/>
7. Dependency Parser or how to find syntactic neighbors of a word <https://towardsdatascience.com/dependency-parser-or-how-to-find-syntactic-neighbours-of-a-word-a9e7d17ffe8>
8. Part-of-speech tagging https://en.wikipedia.org/wiki/Part-of-speech_tagging

Author Info



Srihari V

Srihari has been in Telecom and networking for past 16 years. He has managed various testing teams and has been creating next gen solutions as value adds for leading OEM clients. He is currently a part of the Solutions team and generates AI based solutions to support business needs.



Ramakrishna Narayanan

Ramakrishna Narayanan has been in the field of Artificial Intelligence for the past 2 years. He pursued Masters in IT management and Data Science at University of Texas at Dallas. He has been a part of multiple projects by providing solutions using AI in the field of Testing. He enjoys making people's life easier with AI.



Nikhil Damodaran

Nikhil Damodaran is a Senior Software Engineer with Post Graduate Degree in Data science with two years of experience in leveraging Machine Learning and Deep Learning models to uncover insights into data. He has experience as ML solution developer in speech recognition and NLP projects. Some of his publications include: -

1. "Effect of Decolorized Images In Scene Classification Using Deep Convolution Features."
2. "Scene Classification Using Transfer Learning."
3. "Detection of Duplicates in Quora and Twitter Corpus."

ABOUT HCL TECHNOLOGIES

HCL Technologies (HCL) is a leading global IT services company that helps global enterprises re-imagine and transform their businesses through digital technology transformation. HCL operates out of 44 countries and has consolidated revenues of US\$ 9.3 billion, for quarter ending September, 2019. HCL focuses on providing an integrated portfolio of services underlined by its Mode 1-2-3 growth strategy. Mode 1 encompasses the core services in the areas of Applications, Infrastructure, BPO, and Engineering & R&D services, leveraging DRYICE™ Autonomics to transform clients' business and IT landscape, making them 'lean' and 'agile'. Mode 2 focuses on experience-centric and outcome-oriented, services such as Digital and Analytics Services (BEYONDigital™), IoT WorkS™, Cloud and Security, utilizing DRYICE™ Orchestration to drive business outcomes and enable enterprise digitalization. Mode 3 strategy is ecosystem-driven, creating innovative IP-partnerships to build products and platforms business.

HCL leverages its global network of integrated co-innovation labs, and global delivery capabilities to provide holistic multi-service delivery in key industry verticals including Financial Services, Manufacturing, Telecommunications, Media, Publishing, Entertainment, Retail CPG, Life Sciences Healthcare, Oil & Gas, Energy & Utilities, Travel, Transportation & Logistics and Government. With 120,000 professionals from diverse nationalities, HCL focuses on creating real value for customers by taking 'Relationships Beyond the Contract'. For more information, please visit www.hcltech.com.

ABOUT HCL ENTERPRISE

HCL is a \$8.6 billion leading global technology and IT enterprise comprising two companies listed in India - HCL Technologies and HCL Infosystems. Founded in 1976, HCL is one of India's original IT garage start-ups. A pioneer of modern computing, HCL is a global transformational enterprise today. Its range of offerings includes product engineering, custom & package applications, BPO, IT infrastructure services, IT hardware, systems integration, and distribution of information and communications technology (ICT) products across a wide range of focused industry verticals. The HCL team consists of over 147,000+ ideapreneurs of diverse nationalities, who operate from 44 countries including over 500 points of presence in India. HCL has partnerships with several leading global 1000 firms, including leading IT and technology firms. For more information, please www.hcl.com



www.hcltech.com

Hello there! I am an Ideapreneur. I believe that sustainable business outcomes are driven by relationships nurtured through values like trust, transparency and flexibility. I respect the contract, but believe in going beyond through collaboration, applied innovation and new generation partnership models that put your interest above everything else. Right now 120,000 Ideapreneurs are in a Relationship Beyond the Contract™ with 500 customers in 32 countries.

How can I help you?

HCL