

8-2016

Building an Application for the Automation of On-boarding of an Employee

Sai Kowshik Dandavathi

Follow this and additional works at: https://repository.stcloudstate.edu/mme_etds

Recommended Citation

Dandavathi, Sai Kowshik, "Building an Application for the Automation of On-boarding of an Employee" (2016). *Culminating Projects in Mechanical and Manufacturing Engineering*. 58.
https://repository.stcloudstate.edu/mme_etds/58

This Starred Paper is brought to you for free and open access by the Department of Mechanical and Manufacturing Engineering at theRepository at St. Cloud State. It has been accepted for inclusion in Culminating Projects in Mechanical and Manufacturing Engineering by an authorized administrator of theRepository at St. Cloud State. For more information, please contact rswexelbaum@stcloudstate.edu.

Building an Application for the Automation of On-boarding of an Employee

by

Sai Kowshik Dandavathi

A Starred Paper

Presented to the Graduate Faculty of

St. Cloud State University

in Partial Fulfillment of the Requirements

for the Degree

Master of Engineering Management

August, 2016

Starred Paper Committee:

Hiral Shah, Chairperson

Ben Baliga

Balasubramanian Kasi

Abstract

This project was implemented in a company is a business aligned technology services provider with proven capabilities in software application development, application management and systems and technology integration services and an expert in building business value for global financial services and Fortune 500 companies. This project proposal will focus on implementation of algorithms to automate hiring process. As this project involved collecting information about the candidate, matching the candidate to an open position, scheduling an interview with a hiring manager, collecting and processing the feedback upon which a hiring decision is made, and then extending an offer to the candidate.

Acknowledgments

This project would not be possible without the help and encouragement from the professors of my committee Dr. Hiral Shah, Dr. Ben Baliga, and Prof. Balsy Kasi. The professors really helped me in gaining knowledge in every subject that was taken during my Master's in Engineering Management at St. Cloud State University. I would also like to thank Dr. Ben Baliga. The resources that were provided during the study at St. Cloud State University were tremendous. I would also like to thank Pegasystems for their software and literary bank which helped in simplification of the project. Finally, I would like to thank my friends and family for providing a great support in the fulfillment of my dream study at St. Cloud State University.

Table of Contents

	Page
List of Figures	6
Chapter	
1. Introduction	8
Problem Statement	9
Objective of the Project	9
Nature and Significance of the Problem	9
Project Questions	10
Limitations of the Project	11
Summary	11
2. Background and Review of Literature	12
Introduction	12
Literature Review	12
Methodology	15
Summary	16
3. Methodology	17
Introduction	17
Prologue for Creation of Application	17
Design of Study	20
Budget	38
Timeline	38

	5
Chapter	Page
Summary	39
4. Data Presentation and Analysis	40
Introduction	40
Data Presentation	40
Data Analysis	42
5. Results, Conclusion, and Recommendations	43
Introduction	43
Results	43
Project Questions	43
Conclusion	44
Recommendations	45
References	46

List of Figures

Figure	Page
1. Hr. services application basic details	21
2. Stages of application design	21
3. Adding specific sub groups to the stages	22
4. Types of flow actions in application with instructions, these are important for the basic built of application	23
5. Implementation class of the case	24
6. Stage based case management of onboarding	27
7. Stage based case management of medical plan	28
8. List of data that is collected in medical plan	28
9. Cost incurred after medical plan	29
10. Automation of costs in the application for the different medical plan	30
11. Details collected for the list of beneficiary to successfully elect a beneficiary to the benefits plan	30
12. Sample user interface for the beneficiary enrollment	31
13. Stage based case management of payroll enrollment	31
14. List of details collected in payroll direct deposit enrollment	32
15. Connector properties in the evaluation case	33
16. Basic class hierarchy explaining the classes used for building the application	34

Figure	Page
17. Rule hierarchy explaining the rules that are created for the application	35
18. Data hierarchy explaining the data that is stored in the application	36
19. Supporting specification types	36
20. Declaration of properties	37
21. Project timeline	38
22. Different types of report generated by application	40
23. PAL tool readings	41
24. Performance summary graph of new applications has a total elapsed time of 5.34 sec	42

Chapter 1: Introduction

For the past 20 years, The Company had helped people across the globe balance the interdependency between life and technology. Today, as the industry-defining leader of technology solutions, the company ensures 280 million consumers' devices and appliances stay online and on the job in this fast-moving, tech-driven world. The company help its clients retrieve the seventeenth (and perfect) picture of your dog in that sweater, and make sure the undeliverable email is delivered. The company has the people who know what that one button does. In addition, sometimes what it does not do. The company has always worked as though life depends on us; and, increasingly, it does. Technology is virtually everywhere. It drives performance at work, powers a growing number of tasks at home, and guides us as we move between the two.

The Company is a technology company. It ensures technology and people are harmonious. In addition, your life is in balance. We do so across the digitally connected globe, by speaking six languages, and by working across any device, platform or provider. The Company is unique, and it took two of innovating, and empathizing, to build.

This capstone project proposal focused on implementation of algorithms to automate the on boarding process in its recruiting. As this project involved, the candidate acceptance of the offer and it ran through the first 90 days of employment. A new employee record had to be created, invitation for the welcome kit, the new employee record to be created for the enablement of employee benefits plan, setting

up of orientation, etc., it was tedious to manually perform all these communications between the company and the employer. In addition, the proposal also includes the problem statement, objective, nature, and significance of the problem.

Problem Statement

This company used Applicant tracking system to manage onboarding process. Over time, some business processes were modified in order to fit into the Advantage mold. A new system was developed from the scratch that had enabled them to easily configure/ make changes on the go and efficiently address on boarding services needs in a brisk method. In on boarding, many manual processes included extending an onboarding form for the enrollment of the benefits to the candidate. In addition, the existing legacy applications were unable to meet the modern day demand leading to several issues. As this process was a repetitive one, by implementing this company could easily process the on boarding process of its employees.

Objective of the Project

The objectives of the project were:

- Decrease turnaround time of the onboarding process by 20%.
- Reduce the cost of the overall cycle by 15%.
- Increase in the performance time by 20%.

Nature and Significance of the Problem

The present system had traditionally been thought of as a way to manage the HR functions and employees of a business. As the technology developed and HR functions changed, there was a need to develop software that handled the process of

recruitment and hiring. Having the right recruitment software can change the way you recruit, saved the time and money during the recruiting process, and most importantly, ensured that you are hiring the best people.

The amount of time spent on paper based systems and administrative tasks had caused many HR managers to look carefully at the benefits software.

In another case, the modern day web applications supported cross browser compatible to mobiles and tablets. Our current legacy application has some constraints when it came to this feature. Therefore, there was a need to address this to make it user friendly.

Project Questions

The following project questions were answered after the implementation of automation process.

1. What was the turnaround time that had been reduced in the whole process?
2. What were the total cost savings by this automation process?
3. What was the improvement in the performance by automation process?
4. What was the best way to achieve the compatibility in different kind of devices without reducing the performance?
5. What were the changes to be made in the project implementation to make it a user friendly and subtle one?

Limitations of the Project

This project was designed and tested with the limited test data available at the company. The maximum potential of the project can only be found when tested with the enormous real time data available with the end users of this project. However, every calculation used in this project is of a great accuracy with the data available at the time of the project.

Summary

This chapter briefly covered many aspects of this project prominently to determine the actual problem that exists and how it affects in real time, main motive of the project, list of questions that are going to be answered at the end of the study, basic limitations of the project and, finally, the definition of all the terms that are used in this project to fully understand the meaning of each term. The next chapter covers the literature background knowledge associated with this project.

Chapter 2: Background and Review of Literature

Introduction

This chapter focuses towards reviewing the literature of the problem, literature related to the methodology that has been implemented in the process of solving the problem and the background of Company and the issues related to it.

Literature Review

Agile software development is a gathering of software development systems taking into account iterative and incremental development in which prerequisites and arrangements advance through coordinated effort between self-sorting out, cross-useful groups. It advances versatile arranging, transformative development and conveyance, a period boxed iterative approach, and supports fast and adaptable reaction to change. It is a structure that advances desire tight emphases all through the development cycle (Highsmith, 2001).

Pegasystem develops, markets, licenses, and supports software, which allows organizations to build, deploy, and change enterprise applications. It has a few products that attention on client administration and continuous examination. They are reliably giving superb software items in a perceptive way to the customers by meeting the due dates. They likewise give the right devices, innovations and ability sets to execute most far-reaching and complex undertakings with a great efficiency. Pegasystems clients are in different divisions, for example, financial services, insurance, healthcare, communications and media, life sciences, and government.

Pegasystems Inc. was founded in 1983 by CEO and Chairman Alan Trefler, based in Cambridge, Massachusetts (Pegasystems, n.d.).

Agile methods are advantageous over traditional software development methods when it comes to constant customer participation, customer satisfaction with early project delivery and approval of requirement changes at any stage of software development life cycle. Therefore, the practice of agile methods has been increasing in recent years. Some of the well-known agile methods that are in practice today are Adaptive Software Development, Feature Driven Development, and Scrum, etc. A segregation among available agile methods helped managers who need to introduce agility into their organization by providing a thorough learning on different features and aspects of the agile methods. By segregating different agile methods, managers can easily choose the agile method that best suited the project requirements and organizational characteristics. Different segregation techniques to compare agile methods were introduced in recent years by researchers. We have compared the agile methods Adaptive Software Development and Feature Driven Development in order to evaluate their degree of agility and coverage degree with respect to the two Knowledge Areas, (1) Software Requirements, and (2) Software Construction. The result of our segregation showed that Adaptive Software Development mainly varies from Feature Driven Development, no predefined techniques or specific practices that were using this in Adaptive Software Development. Feature Driven Development had some predefined techniques or customs which are to be followed during software development. The principle of Adaptive Software Development is to focus more on

results rather than steps to achieve that result. A technique which is only implemented if it is necessary to get the best output in Adaptive Software Development (IEEE Xplore Digital Library, 2011).

The Company initially followed Waterfall Methodology which was later changed to Agile Methodology as agile development methodology provides opportunities to assess the direction of a project throughout the development lifecycle. This was achieved through regular cadences of work, known as sprints or iterations, at the end of which teams must present a potentially shippable product increment. By focusing on the repetition of abbreviated work cycles as well as the functional product they yield, agile methodology was described as “iterative” and “incremental.” In waterfall, development team only had one chance to get each aspect of a project right. In an agile paradigm, every aspect of development—requirements, design, etc.—were continually revisited throughout the lifecycle. When the team had project reviews every other week, it had opportunity to steer in the desired direction. This way the project was always under control and wastage of energy and resources was minimized (Highsmith, 2000; Pressman, 2008; Royce, 1970).

The results of this “inspect-and-adapt” approach to development greatly reduce both development costs and time to market. Because the team could develop software at the same time they were gathering requirements, the phenomenon known as “analysis paralysis” is less likely to impede a team from making progress. And because the team’s work cycle was limited to two weeks, it gives stakeholders

recurring opportunities to calibrate releases for success in the real world. Agile development methodology helped company build the right product. Instead of committing to market a piece of software that hasn't even been written yet, agile empowered team to continuously re-plan their release to optimize its value throughout development, allowing them to be as competitive as possible in the marketplace. Development using an agile methodology preserved a product's critical market relevance and ensured team's work did not wind up on a shelf, never released (Agile Methodology, 2008; Royce, 1970).

Methodology

Various tasks which will be executed through the Agile Methodology are:

1. Planning

- Selection.
- Infrastructure Requirements.
- Security related information and gathering.
- Service Level Agreements and its conditions.
- Setting up the orientation and benefits program.

2. Analysis

- Define time frame to the project.
- Resource planning for both Development and QA teams.
- Planning of resources of the company and employee.

3. Design

- Break down of tasks into a smaller sub tasks.

- Test Scenario preparation for each sub task.
- Regression Automation.

4. Execution

- Coding.
- Unit Testing.
- Execution of Manual test scenarios.
- Generate the appropriate reports after each sub task.
- Conversion of Manual to Automation regression test tasks.
- Process review.

5. Closure

- Pilot Launch.
- Training.
- Production Launch.
- SLA Guarantee assurance.
- Review SOA strategy.

Summary

This chapter briefly covered the background and literature of the problem. It also described the methodology used and explained different phases of the Agile methodology.

Chapter 3: Methodology

Introduction

A case had stuff associated with it. It had people assigned to work the case; tasks to be performed; relevant documents and correspondences; a status; a history of actions taken; and a resolution.

Prologue for Creation of Application

Creating case types and stages. The Case type's stages started with stage-based case design by confirming the understanding of case types and stages.

Effective Case Management required that individual contributors' complete assignment in an organized way so they can resolve a case adequately. These undertakings can be seen as sections on a program, speaking to destinations to be finished to determine the case. In its least complex shape, a case sort characterizes the undertakings and choices expected to finish a business transaction. In this product, a case type is a collection of Pega-related artifacts such as data elements, UI screens, processes and sub cases, decisions and integrations used to complete the tasks for a specific case. With no setting, nonetheless, the connections and conditions between these goals are undetectable to both application architects and the general end users (Pega 7, n.d.).

A "stage" is primary step in organizing the different tasks required to complete work associated with a case.

Case management gave us a detailed view of a business transaction. It allowed the business to see, and interact with, the business transaction the way it

moves through the organization. This became especially useful when we considered that a “case” could include not only processes, but also other cases—or, sub cases—as well (Pega 7, n.d.).

Creating primary, alternate and resolution stages. There are three types of stages. These are used for any given case type, they are primary path stages, alternate stages and resolution stages.

“Primary path” stages are stages the case would pass through in a “normal course of events” and are generally free of exceptions.

“Alternate stages” are those stages that are not part of the “normal course of events” but must be available under certain circumstances—or exceptions to the normal course of events.

“Resolution stage” is the third type of step in this discussion which is used to resolve or close a case when primary and alternative steps fail. It is used to terminate the process itself.

Both primary path stages and alternate stages can be defined as a “resolution” stage. Classifying a stage as a “resolution stage” provides a visual indicator that a case can be resolved, or closed, in that particular stage. This can be useful—someone who does not know the process very well can see, and understand that is where the case can come to a close.

A primary path stage can be configured to automatically transition to the next primary stage when all steps in that stage are complete or, set to manually transition to another stage by an end user or as defined in a process model. Alternate and

resolution stages can only ever use a manual transition to a next stage (Pega 7, n.d.).

Creating the application. The Onboarding course started with the acknowledgement of an offer and carried out till the initial 90 days from the start date. The course of action initially was to manually create the employee record before the actual start date. After the record was initiated, an email was to be sent with the information in the initial two days of acknowledgement of the offer.

Prior to the new employee's start date, the hiring manager needs to request resources for the new recruit and begin setting up the new employee's benefit plan. HR should begin setting up the orientation schedule as well.

The first thing to be done after the employee is onboard is to set up Orientation. In the orientation, it was scheduled to explain the employee about the company policies, mandatory and optional courses to be taken by the employee. In the next step of orientation, the company has to ensure that employee has the correct insurance coverage and is accordance with the company policies. It is the duty of the HR department to explain about the various medical plans available with the company and to which an employee can enroll to through the company. This phase of orientation contains the details of various dental, vision and general insurance options available for the employee through the company. The next phase is to guide the employee to the payroll department and setting up the required payroll options like setting up an account for direct deposit etc. The critical step in the process is the employee evaluation on immediate, short term and long-term process.

This process is typically completed in the first 3 months of the hiring depending on the expectations and goals discussed with the manager/mentor. The above steps can be done in single step assignment, multi-step process assignment or a case. In the second iteration of defining the onboarding case type, we identified the following steps (New Employee Onboarding: Best Practices for New Hires–Trello Blog, 2015; Pega 7, n.d.).

- Employee Record
- Send Welcome Kit
- Setup Orientation Schedule
- Request Resources
- Benefits Enrollment
- Payroll Setup
- Evaluate Employee

Design of Study

The study covers the architecture used for the design of the project: Creating New Application and Creating the HR Department Application.

The New Application wizard was used and provided the following information to create an application:

- Basic application information, such as the name and description of the application.
- Business objectives that the application must satisfy.

Field	Value
Application	HR Department
Description	The HR Dept manages Human Resources of Horizon Corporation. Employers can use this application to manage onboarding of an employee
Built on Application	PegaRules
Application Structure	Implementation Only
Organisation	Horizon
Business Objective	Reduce response time and automating the process

Figure 1. Hr. services application basic details. Figure explaining about the hierarchy in which the application was developed.

Identifying Case Stages. Stage Designer was used to add the following four primary stages to your case: Onboarding, Pre-arrival, Arrival, and Evaluation.

Figure 2 shows case stages of application divided in four stages—Onboarding, Pre-arrival, Arrival, and Evaluation.



Figure 2. Stages of application design. Figure explaining about the preliminary stage of developing of application.

Figure 2 explains about the basic stage by case management involved in creating the project. In this case, it explained about the stages of application design.

Adding Steps to a Case Stage. Once a case is divided into stages, the next step is to add steps to each stage. Each step represents a specific action or item to complete in order to process—and ultimately resolve—a case.

Figure 3 shows that each step represents a specific action or item to complete in order to process and ultimately resolve—a case.

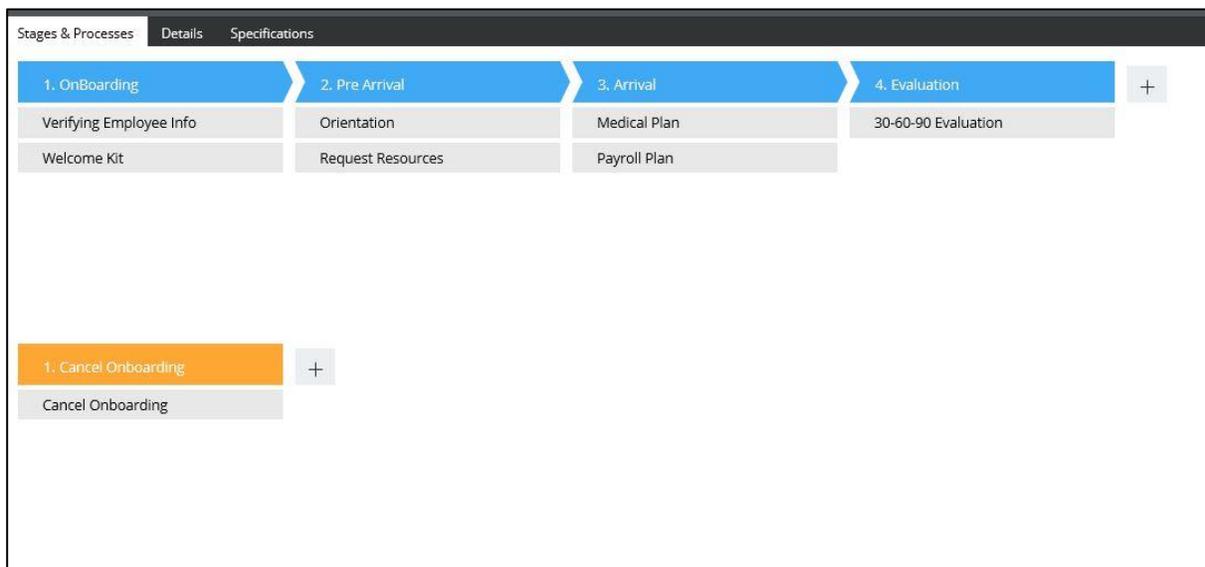


Figure 3. Adding specific sub groups to the stages. Figure stating the basic structure of the application.

Figure 3 explains about the skeleton of the application after the addition of the preliminary stages.

The approach to designing the Onboarding case type was to add the following stages and steps:

Intake

Create Employee Record (Single Step Assignment)

Send Welcome Kit (Single Step Assignment)

Pre-Arrival

Setup Orientation Schedule (Single Step Assignment)

Request Assets (Single Step Assignment – Start step upon stage entry)

Setup Enablement Plan (Single Step Assignment – Start step upon stage entry)

Wait for New Employee Start (Multi Step Process)

Orientation

Benefits Enrollment (Subcase)

Setup Payroll (Subcase–Start step upon stage entry)

30-60-90 Evaluation

Evaluate Employee (Single Step Assignment)

Cancel Onboarding (Alternate Stage)

Cancel Onboarding (Multi Step Process)

Assignment	Flow Action	Instructions
Verify Employee Info	Populate Employee Info	Populate Employee Info from the Recruit system
Orientation	Orientation Schedule	Set Up Orientation for New Recruit
Request resources	Resources	Request Resources for New Recruit
Medical Plan	Medical Plan	Explain about various Medical Plans Available
Payroll Setup	Payroll	Set up Payroll in system
Evaluation	Evaluation	Evaluate the employee

Figure 4. Types of flow actions in application with instructions, these are important for the basic built of application.

Figure 4 explains the flow actions that are involved in creation of the application. The flow action in this application defines about the action that has to be taken when a flow reaches a particular junction.

Each case was further divided into four different sub cases to facilitate the process simplification. Each Subcase is assigned to an Implementation class depending on the kind of information that is being collected and stored. Therefore, we have four different implementations classes one each for onboarding, Medical, Payroll and Evaluation. The data that is being stored and accessed through these are stored in data tables. This process of storing data tables in different data classes helped in keeping the information discrete and confidential among different group in the company. This process is done to safeguard the personal information of the Employee as at any given point of time none of the department had all the information about the candidate as information was only made available on a need basis to different departments.

▼ CASE TYPES

Use the checkbox to include case types in the create menu.

NAME	WORK ID PREFIX	IMPLEMENTATION CLASS	ICON
<input checked="" type="checkbox"/> Onboarding	O-	Hori-HRDept-Work-	🔍 ⚙️
<input checked="" type="checkbox"/> MedicalPlan	M-	Hori-HRDept-Work-	🔍 ⚙️
<input checked="" type="checkbox"/> PayrollSetup	P-	Hori-HRDept-Work-	🔍 ⚙️
<input checked="" type="checkbox"/> Evaluation	E-	Hori-HRDept-Work-	🔍 ⚙️

+

Figure 5. Implementation class of the case. The data is stored at various different classes as mentioned above.

Figure 5 describes the implementation of tasks in the application. The application is divided into various classes for the feasibility of maintenance.

Onboarding. This is the first of the four subcases in the automation of onboarding of an employee. This subcase has three different steps. This process is handled by the Facilities department before the employee start date.

- a) Employee Record: In this step, the employee details have been retrieved from the Horizon-Data-Recruit Record Class and auto-populated into the system with the help of unique identification number given to the New recruit after accepting the offer.

This step contains records of Employee Personal, educational and professional Information.

- b) Set up Employee Resources: Facilities department further gathers the data needed by the new recruit in the department and provides them as directed by the hiring manager of the resource. The UI screen contains the following details. Different teams like facilities, hardware, software, IT, etc. coordinated by the facilities team provide the adequate resources required by the employee.
- c) Welcome kit: This is the final step in the subcase Onboarding where in the facilities team emails the new recruit from within the process containing the details of reporting date, time, manager details and contact details etc. The typical email form is configured in the Email welcome kit task under this step.

Our business policies indicated that the hiring manager should be notified via email when a new employee record is created and when the assignments for requesting the new employee's assets and setting up the new employee's enablement plan were pending.

We also needed to send a welcome letter, which will be mailed to all new employees once the employee record is created. The content of the letter should read:

Dear [employee's full name],

Welcome to the company and the [department name] team! I am delighted you are joining us as a [new employee's job title]. Your role is critical in fulfilling the mission of our department and the Company.

The [department name] team is here to support your transition so, please know that you can call on any of us to assist you. We are looking forward to you joining our team and your success at Company.

Sincerely,

[Manager's name]

This process is automated depending on the type of the recruit, full time employee and part-time. This automation is done with the help of the Flow diagram, which is being in turn called by the flow action within the process. This flow is customized to suit the different requirements of the new recruit. This customization was done with the help of decision shape where the two connectors are branched and an appropriate welcome kit is sent to the new recruit depending on the type of the employee.

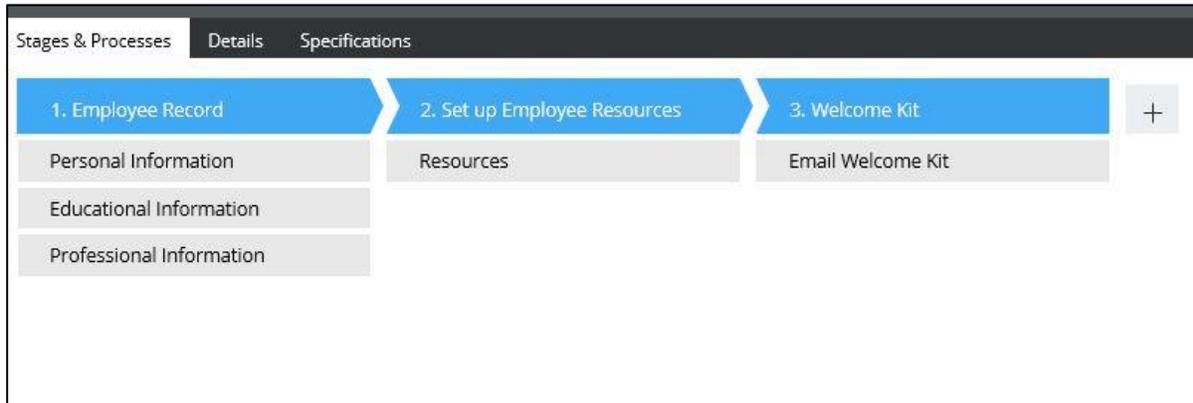


Figure 6. Stage based case management of onboarding. The figure shows the division of application in the case types.

Figure 6 explains about the first case in the application design where employee's personal details like educational information, professional information are taken.

Medical plan: This is the next subtype in the automation of onboarding of an employee process is election of the different medical plans. This stage is divided into three different tasks depending on the nature of benefits required like, general, dental and Vision.

Each task had two different types Basic and Premium/Advanced depending on the deductibles. For basic plan, \$250 cut whereas premium plan had no deductible. These three different types of Medical plan as categorized into three different tasks based on the type of organ that needed to be insured as these three types of insurance is provided by three different third parties but they are together billed to the company.

This Subcase has an additional process that is being coded to add family members of the employee to cover in the medical plan as well. This step has

repeating grid in the User Interface layout to facilitate the addition of any number of qualified family members.

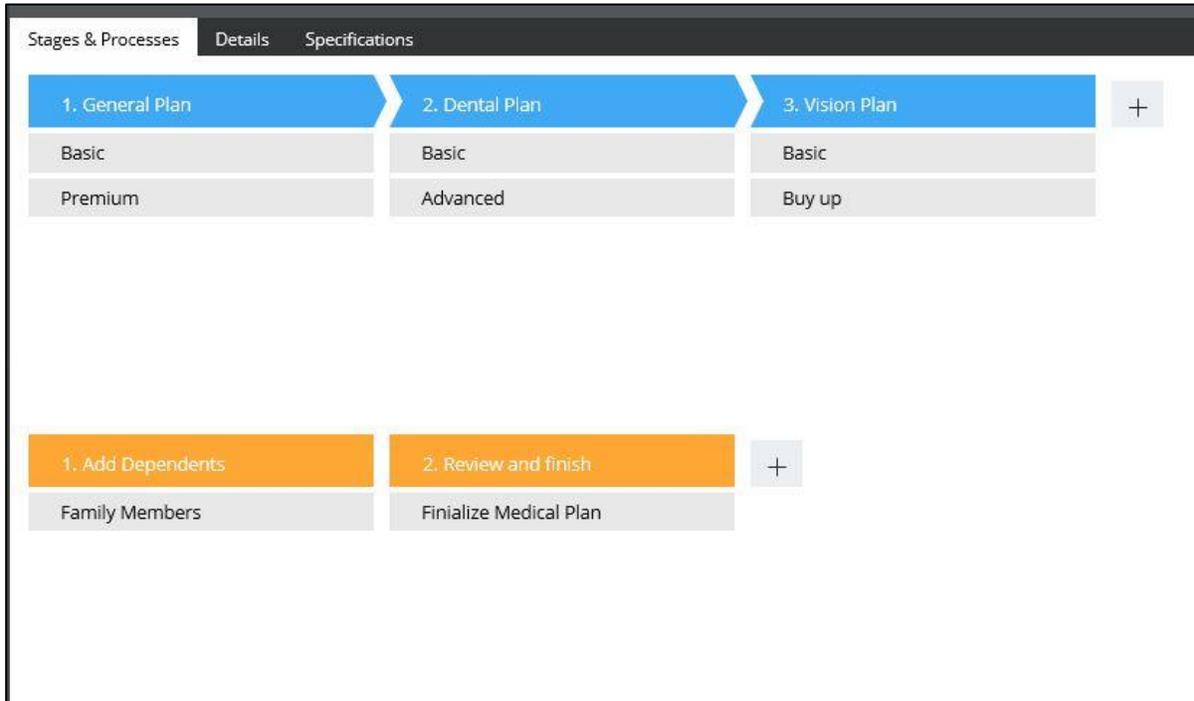


Figure 7. Stage based case management of medical plan. The figure explains how each stage in medical plan is divided into different plans.

Benefits Enrollment	
Stage	Steps
Personal Information	Confirm Personal Information
Benefit Election	Select Medical Coverage
	Select Dental Coverage
	Select Vision Coverage
Dependent Information	Add Dependents
Review and Finish	Confirm Benefit Elections
	Confirm Dependent Coverage

Figure 8. List of data that is collected in medical plan.

Figure 8 shows the checklist to enroll successfully into the Benefits plan.

Our approach to designing the Benefits Enrollment case type was to add the following stages and steps:

Selection

Select Medical Coverage (Single Step Assignment)

Select Dental Coverage (Single Step Assignment)

Select Vision Coverage (Single Step Assignment)

Completion

Finalize Enrollment

These plans were split into three tasks but contain a similar User Interface that contains the following details listed in the table below.

Name	Benefits	Cost
Health Medical	Basic Medical	20.96
Health Dental	Dental Coverage	4
Health Vision	Vision Coverage	5.6
Basic Group Life	Group Life Insurance	3.5
Basic AD&D	Basic Employee AD&D	4.2
Supplemental Group Life	Supplemental Insurance	7.6
Short Term Disability	STD Plans	16.73
Long Term Disability	LTD Plans	5.73
Life Lock Plans	Employee benefit from the Company	9
		77.32

Figure 9. Cost incurred after medical plan.

Figure 9 is an example for the basic plan with various selections done in a basic format.

Figure 10. Automation of costs in the application for the different medical plan.

Figure 10 explains about the UI of the Medical plans that are selected accordingly to the elected options.

In addition to this screen, the employee has to fill out a form indicating the beneficiaries to which the insurance benefits have to be credited in case of any unforeseen circumstances. The UI contains the below details.

Manage Beneficiary	Details
First Name	
Last Name	
Date of Birth	
Relationship	

Figure 11. Details collected for the list of beneficiary to successfully elect a beneficiary to the benefits plan.

Figure 11 explains about the essential details that has to be collected to add a beneficiary. This step is mandatory to add a beneficiary of the medical liabilities.

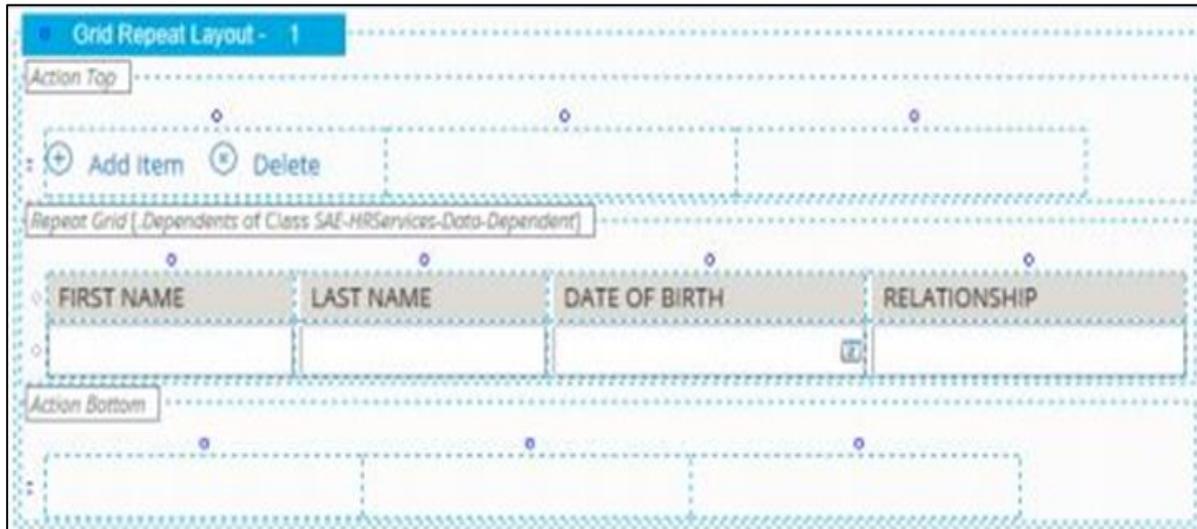


Figure 12. Sample user interface for the beneficiary enrollment.

Payroll set up. This is the next step that had been followed while developing the application for automation of onboarding.

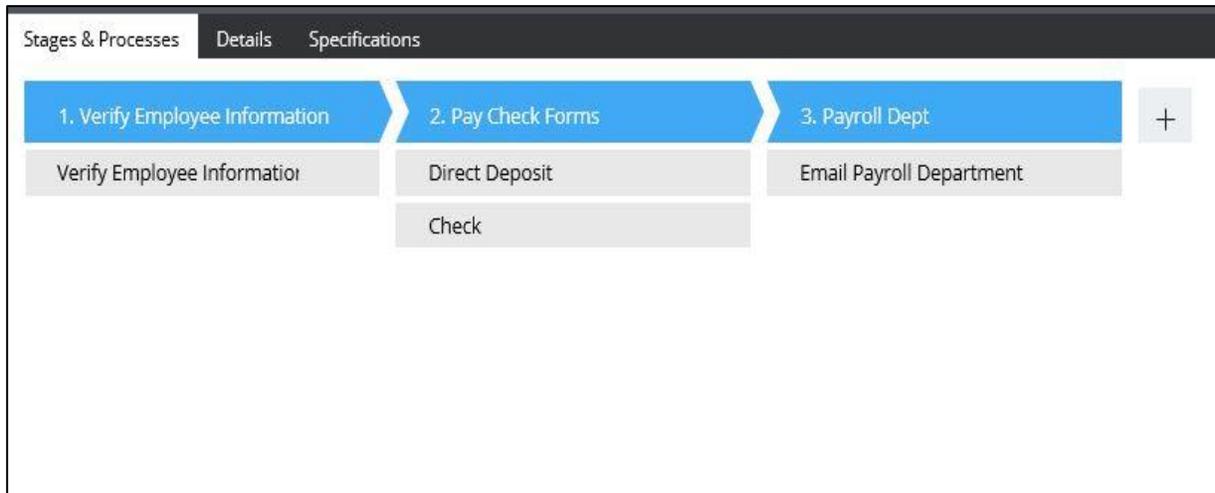


Figure 13. Stage based case management of payroll enrollment.

Figure 13 explains about the case based stage management of the payroll case. This figure shows its details to be collected for payroll case to execute successfully.

This case contains three different tasks.

- a) Verification of Information: This step primarily consists of the verification of employee details that are auto populated with the help of Horizon-HRDept-Data- Class. The employee needs to verify the information that has been populated and proceed to next step if everything is perfect.
- b) Pay Check Forms: This is the crucial step in the Payroll Setup step as this contains the User Interface field with options of direct deposit and Check form. The employee has to choose either of them in order to receive his payroll. This payroll is maintained by a third party site but the information has to be uploaded and sent to payroll department to facilitate the process.

If the employee has chosen direct deposit the User Interface contains the following information.

Index	List
Name of Bank	
Routing number	
Account Number	
Percentage of Pay that has to deposited	

Figure 14. List of details collected in payroll direct deposit enrollment.

In addition to this, the employee has been given choice of adding a maximum of three different bank accounts, for this a repeating grid was used to facilitate the same.

Evaluation: This is the final sub case in the automation of employee onboarding and has a 30-60-90-day evaluation cycle.

Employee has been evaluated on the skills like performance, interpersonal, technical, behavioral etc. and a report has been generated for every 30 days and sent to the manager.

Depending on the review received a flow has been created to discuss the performance of the candidate. The Flow contains the following details.

Connector	Type	Likelihood	Audit Note
Different Position	Result	25	Candidate fits a different position within the organization
Reject	Result	30	Candidate doesn't fit desired position
Promote	Result	75	Promote candidate to different Position
Yes	Result	65	Candidate meets expectation
No	Result	35	Candidate doesn't meet expectations

Figure 15. Connector properties in the evaluation case.

Figure 15 explains about the connectors that connect other assignment in the flow. These connectors contain flow action that does the assignment in the application

Class structure: The figure below shows the class hierarchy in the company. The class hierarchy clearly shows the data that the application has different subclasses for the effective handling of the case.

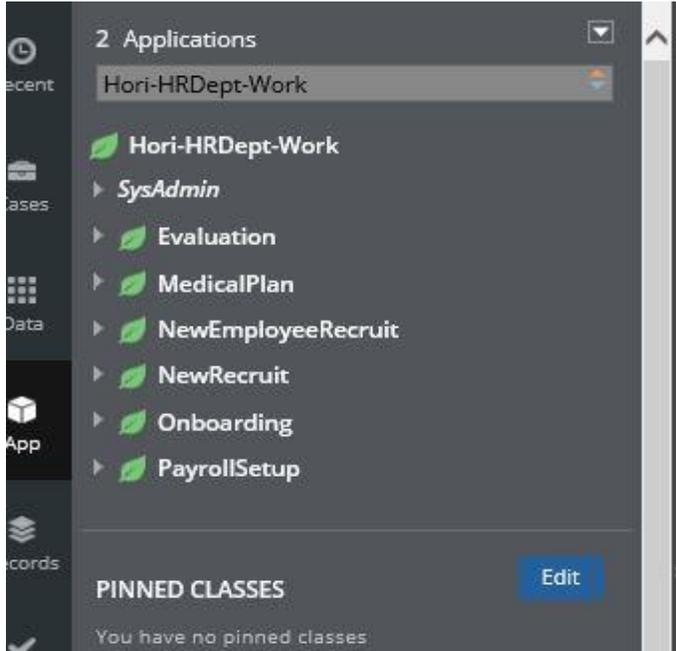


Figure 16. Basic class hierarchy explaining the classes used for building the application.

Ruleset versions: The existing application is built on the following hierarchy with the parent PegaRULES on the top of the list to the HorizonInt 01-01 at the bottom.

Ruleset hierarchy is designed in the way that PEGARules has the most generic rule and it bottoms down to HorizonInt containing the more specific ones. Generic rules can be inherited in the subsequent derivative subclasses whereas the vice versa is not true.

BUILT ON APPLICATION	
Name	Version
PegaRULES	07.10

RULESETS	
APPLICATION RULESETS	
1	HRDept:01-01
2	HRDeptInt:01-01
3	Service
4	CCRApp
5	Hori:01-01
6	HoriInt:01-01

Figure 17: Rule hierarchy explaining the rules that are created for the application.

Data collection and storage: As the application was used by the limited number of agents available in the development center, there was only a little data when the user acceptance test was done. The data collected in the respective Data Classes as Data tables.

The data received through Employer Data is Stored at Hori-Data-EmployeeData

The data received through Evaluation Data is Stored at Hori-Data-EvaluationData

The data received through MedicalPlan Data is Stored at Hori-Data-MedicalPlanData

The data received through Payroll Data is Stored at Hori-Data-PayrollData

DISPLAY NAME	ID	
Employee Data	Hori-Data-EmployeeData	X
Evaluation Data	Hori-Data-EvaluationData	X
Medical Plan Data	Hori-Data-MedicalPlanData	X
Payroll Data	Hori-Data-PayrollData	X

Figure 18. Data hierarchy explaining the data that is stored in the application.

Figure 18 explains about data storage handled in the application. The data is essential to be handled to make an effectively.

The data stored facilitates in the effective storage of data in the System and can be derived easily through the DataTables.

NAME	DESCRIPTION	
Common	Specifications shared by multiple case types	X
Security	Specifications dealing with authentication, authorization	X
Navigation	Specifications specific to getting between case types and gene	X
DataManagement	Specifications specific to managing lists of products, reference	X
SysAdmin	Specifications specific to managing operator profiles, calendar	X

Figure 19. Supporting specification types.

Figure 19 explains about the different types of specification types. These types segregate the application according to its specification so it can be handled effectively.

Defining properties: The next phase followed while creating the application is optimization and declaration of properties associated with the property and storing them in Horizon-Data-RecruitData class as this class can be inherited by all the four above data classes.

The properties declared in the data class were as follows.

Property Name	Label	String Type	Mode
DateOfBirth	Date of Birth	Date	Single Value
Department	Department	Text	Single Value
Dependents	Dependents	text	Page List
EmployeeID	Employee ID	Text	Single Value
FullTime	Full Time	True/false	Single Value
HireDate	Hire Date	Date	Single Value
JobDescription	Job Description	Text	Single Value
JobTitle	Job title	Text	Single Value
ManagerEmail	Manager Email	Text	Single Value
MaritalStatus	Marital Status	Text	Single Value
ReportingManager	Reporting Manager	Text	Single Value
FirstName	First Name	Text	Single Value
LastName	Last Name	Text	Single Value
Relationship	Relationship	Text	Single Value

Figure 20. Declaration of properties. These are essential for the basic inputs used in application.

Sprint cycle outline.

Iteration Structure:

- Sprints will run for duration of two weeks.
- Backlog Refinement will be every Tuesday and Friday at 09:30 a.m.
- Retro and iteration planning will be every Friday at 10:30 a.m.
- Tasking will immediately follow iteration planning at 11:00 a.m.

- A public demo will be held at the end of each sprint at 11:30 a.m. every Wednesday.
- Stories will be considered “Done” when they have been approved by the project owner, Business owner and an appropriate Change Management Request was approved for staging environment.

Deployment Locations.

- All development work will be done on local developer environments and test.
- All development work will be deployed to QA when ready for certification.
- Staging will be used for all feature/update reviews and presentation to Stakeholders.
- No feature/update branches will go to production without project owner review on staging.

Budget

The entire project is completed within the cost provided by the industry.

No extra costs were incurred during the development of this project.

Timeline

Activity	Timeline	Comments
Literature review Proposal	January 2016	
Planning & Requirement Analysis	January 2016	
Designing & Building	February 2016 – March 2016	
Testing	March 2016 – May 2016	
Production	June 2016	
Final Defense presentation	July 2016	

Figure 21. Project timeline.

Summary

This chapter essentially dealt with the Class Hierarchy, rule hierarchy and Property definitions. It also dealt with the various stages followed to build the application, data collection and storage. It also dealt with the Data classes.

Chapter 4: Data Presentation and Analysis

Introduction

This chapter will primarily focus on data representation strategies, and data analysis. Data representation uses a strategy of displaying the data in terms of rows and columns. Data analysis will implement the analyzing the data for validity.

Data Presentation

Out of the box custom reports are generated by Pega to analyze performance, quality, monitor assignments, processes, etc.

Figure 22 shows that Pega by default generates reports that can be readily used by users which are in turn divided into several public categories like performance, quality etc. One can easily develop custom reports that were under private categories.

The screenshot displays the 'Reporting: Access' interface. At the top, there are navigation buttons for 'Refresh', 'Help', and 'Close'. Below this is a 'Report Browser' section with the text 'Showing all reports' and buttons for 'Add report', 'Add category', and 'Help'. A search bar contains 'Recent reports: All reports' with a '12' count and a 'Next' button. Below the search bar is a table of reports with columns for 'TITLE' and 'EMAIL NOTIFICATION'. The table lists various reports such as 'Arrivals Past Eight Weeks SV', 'Average Duration and Timelines by Assignment Type and Action', 'Average duration per flow task', 'Average elapsed time per status', 'Average Performance Time By Task', 'Average Processing Time in Hours By Task and Flow Action', 'Daily Arrivals SV', 'Effort by organization unit', 'Effort in days by Customer', 'Elapsed Status Time Trend', 'External Assignments by Operator', 'List of processes entered by operator', 'List of processes resolved by the current operator', 'Performance Detail in average hours', and 'Performance step detail in average hours'. To the right of the table are two panels: 'PRIVATE CATEGORIES' with 'No Items' and 'PUBLIC CATEGORIES' with a list of categories: 'Analyze Performance (7)', 'Analyze Quality (8)', 'Monitor Assignments (5)', and 'Monitor Processes (11)'. Each category has a gear icon next to it.

Figure 22. Different types of report generated by application.

Below is one of the reports to show the Average duration and timelines by Assignment Type and Action of all cases.

Performance analysis. Results below show the total performance of the application that was out of the box tool.

REQUESTOR PERFORMANCE ANALYZER																					
<input type="button" value="Add Reading"/> <input type="button" value="Add Reading with Clipboard Size"/>																					
INT #	INT COUNT	TOTAL ELAPSED	RA ELAPSED	RULE ELAPSED	I/O ELAPSED	RDB I/O ELAPSED	CONNECT ELAPSED	OTHER I/O ELAPSED	TOTAL CPU	RA CPU	RULE CPU	OTHER CPU	RA COUNT	RULE COUNT	TOTAL RULES USED	ACTIVITY COUNT	RDB I/O COUNT	CONNECT COUNT	OTHER I/O COUNT	ALERT COUNT	TOTAL BYTES
FULL	157	157	192.36	31.60	13.46	43.90	4.90	21.89	49.95	0.00	0.00	0.00	68	1,862	51,932	4,423	1,101	1	1,941	66	9,064,424
INIT	157	157	192.36	31.60	13.46	43.90	4.90	21.89	49.95	0.00	0.00	0.00	68	1,862	51,932	4,423	1,101	1	1,941	66	9,064,424
<input type="button" value="Reset Data"/> <input type="button" value="Save Data"/>																					
BACKGROUND PERFORMANCE ANALYZER																					

Figure 23. PAL tool readings.

Figure 23 explains about the PAL readings taken during the regression testing of the process. The figure has various fields that are being collected but most important of these is total cpu time taken.

PAL Tool Readings analyses the performance of the application and it is out of the box tool.

The total time elapsed by CPU here is 49.95 sec

Performance Summary of legacy application has a total elapsed time of 49.95 sec.

Performance Summary Graph of New Application

REQUESTOR PERFORMANCE ANALYZER																				
Add Reading		Add Reading with Clipboard Size																		
FULL	398	398	44.28	10.86	1.21	4.31	0.00	4.74	23.67	0.00	0.00	0.00	17	3,653	127,866	5,909	462	0	960	38 28,505,723
INT #	INT COUNT	TOTAL ELAPSED	RA ELAPSED	RULE I/O ELAPSED	RDB I/O ELAPSED	CONNECT ELAPSED	OTHER I/O ELAPSED	TOTAL CPU	RA CPU	RULE CPU	OTHER CPU	RA COUNT	RULE COUNT	TOTAL RULES USED	ACTIVITY COUNT	RDB I/O COUNT	CONNECT COUNT	OTHER I/O COUNT	ALERT COUNT	TOTAL BYTES
NIT	398	398	44.28	10.86	1.21	4.31	0.00	4.74	23.67	0.00	0.00	0.00	17	3,653	127,866	5,909	462	0	960	38 28,505,723
Reset Data		Save Data																		

Figure 24. Performance summary graph of new applications has a total elapsed time of 5.34 sec.

The average of the total lapsed time 1 = 49.95 sec

The average of the total lapsed time 2 = 23.67 sec

The total elapsed time difference = 26.28 sec

Percentage of time difference = 53% (approx.)

This was because of the manual process involved with the transfer of cases.

Data Analysis

Based on the above performance analysis time had been decreased by almost 53% because of automation process.

Chapter 5: Results, Conclusion, and Recommendations

Introduction

This chapter will focus on the project results that are obtained after the implementation of the project, conclusion of the project and finally the recommendations that are considered during the implementation of the project.

Results

Agile methodology was implemented for the automation of hiring process. The results obtained are accurate and precisely define.

Project Questions

1. What was the turnaround time that had been reduced in the whole process?

Manual Program: The average of the total lapsed time = 49.95 sec

Automation Program: The average of the total lapsed time = 23.67 sec

The time saved by automation process is 26.28 sec

2. What were the total cost savings by this automation process?

Cost is one of the important factors that need to be considered in any industry. By implementing this automation, company saved a considerable amount of cost.

Manual Program: \$4,800 per month (1 hours per day (3 directors) x 20 days' x \$80 per hour).

Automation Program: \$0 per month

Cost savings: \$4,800 per month.

3. What was the improvement in the performance by automation process?

The target was to increase the performance by at least 20%.

Manual Program: Total elapsed time which is used as a parameter to test the performance = 49.95 sec.

Automation Program: Total elapsed time which is used as a parameter to test the performance = 23.67 sec.

Improvement in performance = 53%.

4. What was the best way to achieve the compatibility in different kind of devices without reducing the performance?

Pega 7 has Inbuilt cross browser compatibility.

5. What were the changes to be made in the project implementation to make it a user friendly and subtle one?

Use of Pega makes the user interface dynamic which makes is user friendly and compatible in any device.

Conclusion

The end users were satisfied by the automation of enrollment, with the kind of reactivity and cross browser compatibility; this has become a most successful project the company has ever taken up. With the reduction of cost with automation, this project found a scope to be expanded across the industry in various other departments as well.

This study was about automating enrollment process using Agile methodology. The company tried to accomplish automatic decision making, send

correspondence, and improving the reports. This project was mainly implemented to overcome the issues of time, cost, scalability, and flexibility in changing assignment logic.

Recommendations

Despite the fact that the task was effectively actualized, a few hazy areas required consideration. Following are a few of them:

- Any logic changes in application needed code change and maintenance.
- As it licensed product one should be very careful in planning and implementation.

References

- Agile Methodology. (2008). *Understanding agile methodology*. Retrieved from <http://agilemethodology.org/>.
- Highsmith, J. (2001). *History: The agile manifesto*. Retrieved from <http://agilemanifesto.org/history.html>.
- Highsmith, J. A. (2000). *A collaborative approach to managing complex systems*. New York: Dorset House Publishing.
- IEEE Xplore Digital Library. (2011, December 22). *Comparison between adaptive software development and feature driven development*. Retrieved June 5, 2016, from <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6175418>.
- New employee onboarding: Best practices for new hires—Trello blog. (2015). Retrieved June/July, 2016, from <http://blog.trello.com/new-employee-onboarding-best-practices-for-new-hires/>.
- Pega 7. (n.d.). Retrieved July 5, 2016, from <https://academy.pega.com/prweb/PegaSSO>.
- Pegasystems. (n.d.). NTT America, Inc. Company. Retrieved from pdn.pega.com.
- Pressman, R. S. (2008). Software engineering. *A Practitioner's Approach*, 6(2), 20-24.
- Royce, W. W. (1970). Managing the development of large software systems. In *Proceedings of IEEE Wescon*, pp. 382-338. Retrieved from

http://leadinganswers.typepad.com/leading_answers/files/original_waterfall_per_winston_royce.pdf.