International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 8, Issue 8, August-2023, http://ijmec.com/, ISSN: 2456-4265

CYCLE TIME OPTIMIZATION USING HIGH MATURITY IMPLEMENTATION FOR TESTING GROUP

Dr. Bhargav Gangadhara,

[1]*Senior Technical lead/Director, Jack Henry and Associates, USA*

[1] *bhargavmtechmem@gmail.com,*

Abstract— This paper explains about resource planning for release to downstream users and cycle time reduction with the aid of an automated tool. Thereby understanding the co- relation between the variables like number of resources, number of request and type of testing, statistical modeling will help to optimize the resource planning.

The automation tool developed instead of manual testing of test cases using the output of statistical modeling will help in work allocation to members of testing team and also to collect the actual time spent on each of the activities performed by them for further analysis. This will also help to provide correct status to development leads. So that results can be communicated to development leads in advance whether their request can be met or not. Resource planning is also required to do random testing in case there are few requests or quality of the product is critical and this will help in identifying how much testing is done.

The next task after resources planning is server upload, which takes a lot of bandwidth of testing group to upload the document and source code. Optimization of server based on understanding of upload sequencing and server load. To meet the current load which will save several million dollars as the server cost is high.

Keywords- *Automation, Co-relation, Testing Statistical modeling, Resource planning, server upload, Upload sequencing, Bandwidth*

1. INTRODUCTION

The organization is into business of providing the solution to mobile phones. This consists of development team, internal testing team and validation team. The testing group has majorly two tasks, one is testing and the other is to release to downstream user. The team is provided with finite number of resources. But there's number of request from development team, which is variable and does not follow any pattern.

Resource planning was first employed by research and analysis firm Gartner Group in 1990 as an extension of MRP I & II (Material Requirements Planning; later manufacturing resource planning) and CIM (Computer Integrated Manufacturing), and the combination of these gave rise to a single term resource planning.

Resource planning systems are often incorrectly called –back office systems", indicating that customers and the general public are not directly involved This is contrasted with ―front office systems" like customer relationship management (CRM) systems that deal directly with the customers or the eBusiness systems such as e-Commerce, e- Government, e-Telecom, and e-Finance, or supplier relationship management (SRM) systems.

Resource planning system is an integrated computer-based application used to manage internal and external resources, including tangible assets, financial resources, materials, and human resources. Its

DOI: https://doi-ds.org/doilink/12.2023-34719133

purpose is to facilitate the flow of information between all business functions inside the boundaries of the organization and manage the connections to outside stakeholders.

2. INTERNAL FLOW OF DATA IN ORGANIZATION

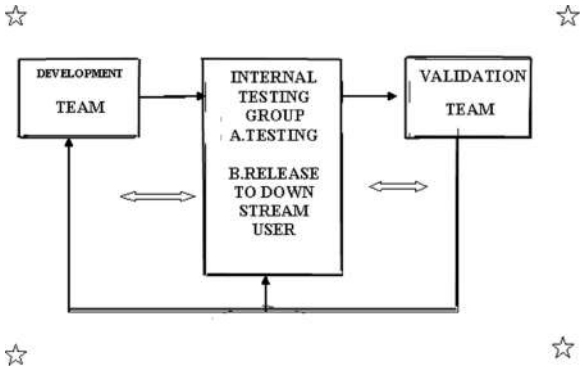

Fig 1: Internal organization structure

The development team release information regarding the time taken for start and end of a model development.

The internal testing group perform the various types of tests, here its sanity test and release the output to downstream users.

The validation team then validates the tests performed and certain control measures are specified as feedback to development leads.

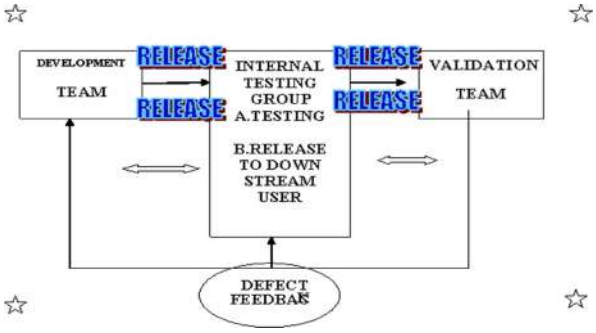

Fig 2: Internal flow of data in organization

The tasks performed for particular activities are:-

- Actual activity that need to be executed.
- Resources will be allocated to these tasks.
- Effort and schedule will be applied on these tasks.
- Dependencies can also be applied.

DOI: https://doi-ds.org/doilink/12.2023-34719133

2.1. To be considered as a resource planning system, a software package should have the following traits:

- An integrated system that operates in (next to) real time, without relying on periodic batch updates.
- A common database that is accessed by all applications, preventing redundant data and multiple data definitions.
- A consistent look and feel throughout each module (sales, manufacturing, accounting etc.).
- The ability to access the system without specialist integration by the Information System (or IT) department.

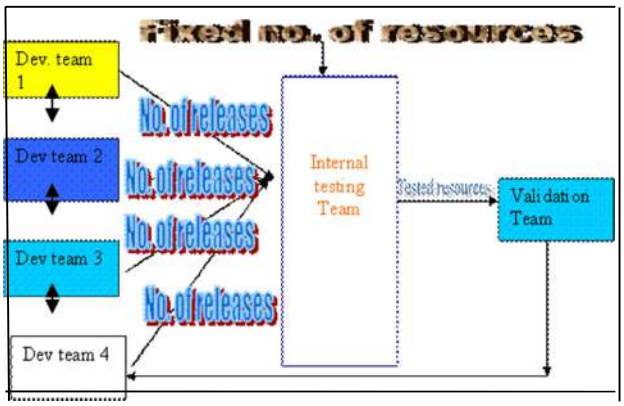

Fig 3: Real time release of data to various teams.

2.2. Manual v/s Automation.

Nowadays manually executing the processes takes more time and the human resources required for the verification of test cases is humongous. Even though the internal testing team works in different shifts , manual deployment of test cases takes more time the remedy for such bottle necks is to automate a tool which helps the testing of functional like sanity testing and non-functional like setting alarm, deleting messages, stress checking the contact capacity which may be limited to 250 contacts only. Here the automation is essential in non-functional test cases where the test engineer can test certain applications such as stress checking any application for more the one day ,even if he's not in working place he can run the automation tool which helps him to stress check that particular application and if there is any critical issue related to test cases then the development lead can be intimated which helps the development leads to know the error well in advance and control measures can be taken to overcome this problems. But in case of manual testing for functional applications like the sanity test companies prefer manual testing itself as knowledge of different domain is essential for a tester which cannot be updated in an automation tool. This untouched area wherein manual testing was essential, an automation tool is developed which reduces the burden of the tester and also manual vigilance is applicable.

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

Those who will be associated with test automation are:-

1. Test Engineers
2. Project Leaders
3. Project Managers

Here we use resource planning and cycle time reduction in testing group. The types of test performed are Sanity Testing or detailed testing...etc. is a brief test of major functional elements of a piece of software to determine if it's basically operational. Sanity testing is used to verify that the software build is ready for System Testing (or more extensive testing). The tests target the most frequently used functional areas, key functions & newly implemented functions for the particular build version.

If an application does not pass the Sanity Test, then further testing activities are suspended.

The conditions which get checked in this test is general, like existence of application, loads without any problem, does

not terminate abruptly, connects to correct database, all menu options are displayed correctly and the start form / page / window is loaded correctly.

2.3. Upload sequencing and server load.

Finally, the next task after automation tool for test cases is to release to downstream users, which takes a lot of bandwidth of testing group to upload the document and source code. Optimization of server based on understanding of upload sequencing and server load. To meet the current load which will save several million dollars as the server cost is high.

Resource planning is also required to do random testing or ad hoc testing in test case (e.g. Sanity testing, Unit testing, integration testing, and system testing and acceptance testing).random testing or ad-hoc testing is done by test engineers wherein they want to test applications such as word dictionary in mobile phones, after all the major testing is done suddenly conducting an ad-hoc test in case there are few requests or quality of the product is critical will improve the quality of the applications in the mobile phone .In turn the customer is satisfied.

3. ROOT CAUSE AND PROBLEM DEFINITION

The resources available from the development team will sent to the internal testing team as explained arlier , the risk involved in process is to be analyzed and the root cause for the risk is to be found.

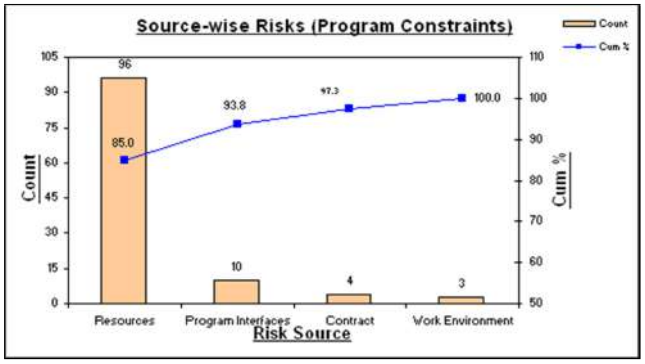

Fig 4:Source-wise Risks (Program constraint)

Here we can clearly see that resource allocation has the maximum risk counts and is causing the problem. The root cause for why the risk cont is high during resource allocation needs analyzed

While going through company processes and procedure.

The one of the criteria found was the manual deployment and release to downstream users, which takes a lot of time. The other criteria is resource planning and work allocation to members of testing group is also a major issue .The cycle time of the above mentioned bottle necks is to be reduced. The next section describes how actually the process takes place.

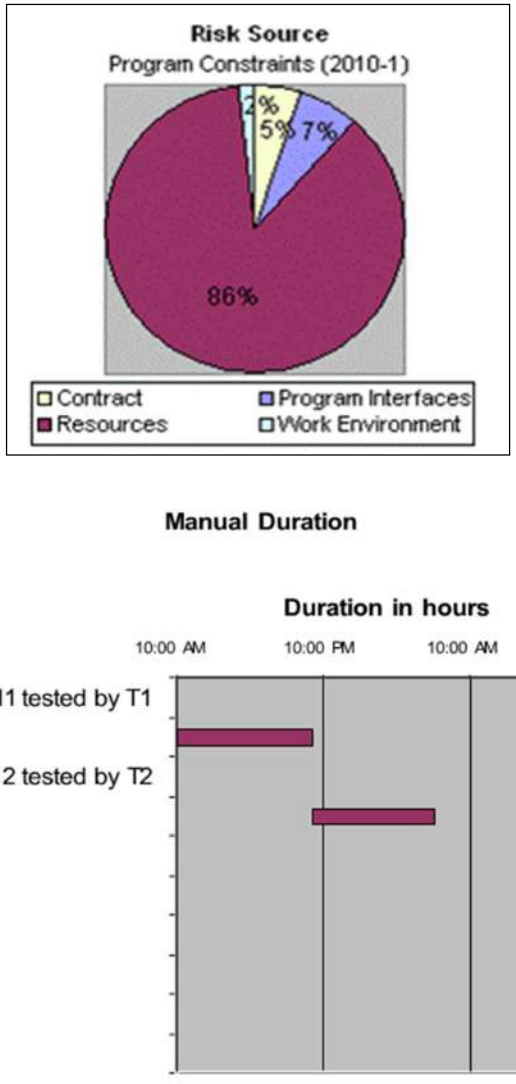

Figure 5: Gantt chart depicting low releases

Root cause 1: manual testing for test cases.

Root cause2: upload sequencing, server load and analysis of bottle necks in networks.

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

Problem statement with Details

Define the problem

- 20 releases per day.
- Scope is to pursue sanity, PRI, key press, Key Map.
- Covering all these things with existing resource is challenging task

Identify where the problem is appearing

- IQM needs to test manually which requires lot of effort.

Describe the size of the problem

- Sanity – 4 hours
- PRI- ½ hours per profile , 270 profile
- Key press – ½ hours per language,35 language
- Key Map- 1 hour per model

Describe the impact the problem is having on the organization

- Delay in release to factory
- Increase VQM rejection

Preventive actions

Root cause 1: automation tool instead of using manual testing

Root cause 2: optimize the overall cycle time of server based on understanding of upload sequencing, server load and analysis of bottle necks in networks.

DATA COLLECTION

4.1. Raw data

The data collected from March to October from the respective development leads and with the help of simular software and also manually analyzed the given metrics, also prepared a line chart for number of models-months taken for developing particular models. The data collected from the model development leads are shown below.

Month	No of models
Mar	3
Apr	4
May	9
June	13
July	16
Aug	16

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

Sept	21
Oct	23

Table 1: Development of models per month

The development of the models and duration for development is also verified and for better understanding line charts where created as shown in Fig 8.

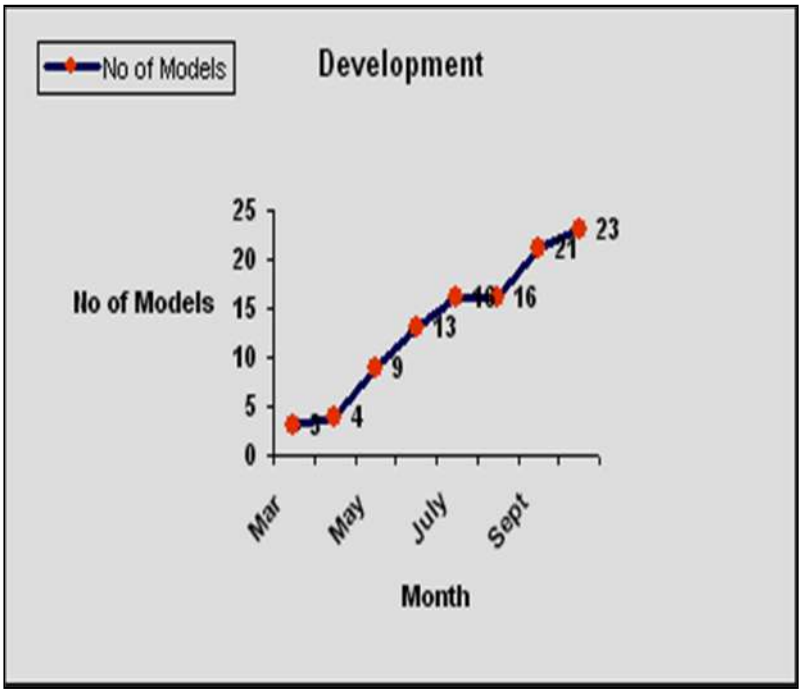

Fig 6: Line chart for development No. of Models-months

The Deployment of test cases data were also collected from model leads and analyzed the variance in the particular data per month. Sanity test which is for functional application such as key press, key map were collected and the time taken for each test conducted are shown below in Table 2. As we know that the functioning of keys in a mobile phone is a major criterion so the testing team usually conducts such tests frequently. Here the test leads conduct sanity test and the time management system which is an internal system in the organization calculate the time taken for each test and also time taken for each activity is also noted such as key map ,key press and any special applications such as word dictionary. The Table 2 clearly shows the test conducted and the time taken for each month. After the further analysis of data for future upcoming months.

DOI: https://doi-ds.org/doilink/12.2023-34719133

	Deployment			
Month	Sanity test	Key Map	Key press	Review
July	1032	491	435	9276
Aug	516	158	7711	14275
Sep	3928	327	2911	10317
Oct	24900	3233	0	14749

Table2.Deployment of sanity test

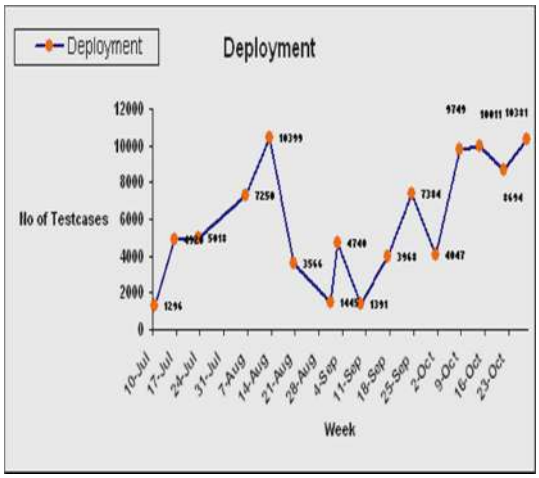

Fig 7: Deployment of No. of test cases-week

4.2. ANALYSIS AND CONTROL

The ranges given in the acceptance line can be used to determine whether the performance of review falls within acceptable limits. To ensure that the performance of a review is checked with respect to control limits after each review takes place , this check is specified with respect to the control limits after each review takes place, this check is specified as an exit criterion for the review process Although the exit criteria can be defined as in-range checking of all of the various parameters, because defects is the central purpose of reviews, the exit criterion is the overall defect density should lie within the specified limits (Checking that the defects densities for the two types of defects are within the appropriate ranges can also be used.)If the number of defects found during the review is within the range given in the acceptance line, the review is considered effective, the exit criteria are satisfied, and no further action is needed for this review.

If the density of defects found in a review is not within the range given in the capability acceptance line, then the exit criteria are not satisfied. In software, however, the fact that the performance has gone out of the specified range does not automatically mean process failure. Instead, the moderator has to critically evaluate the situation and decide on the next steps. The preparation rate and review rate becomes very useful here—if the review rate is –too fast‖ as compared with that given in the acceptance line, then the reason for minor and critical defects can also be useful in this analysis. When the critical process is

DOI: https://doi-ds.org/doilink/12.2023-34719133

recognized we try to find out the root cause of the critical issue as explained earlier.

To solve the first root cause in a process

The comparison of manual deployment of test cases like the functional and non functional test cases which is comp aired with the automated tool evaluated test cases .Here we can see a drastic reduction in time e.g. - if a manually tested test case takes one hour then the automated tool tested test case takes three minutes, it's an astonishing 57 minutes difference between the manual and automated test case.

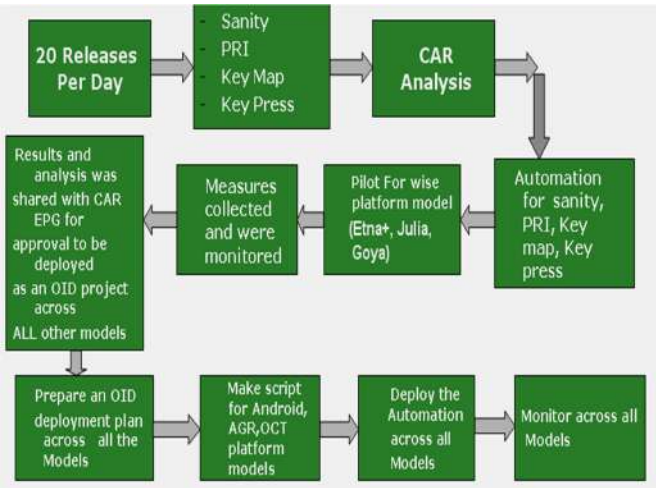

Fig 8: Flow diagram of root cause 1 events

Finally to solve the problem relating to release to downstream user which caused serve down issues, it is necessary to understand the co-relation between the variables like

i. Number of resources,

ii. Number of request and

iii. Type of testing.

Statistical modeling will help to optimize the resource planning. As shown in the Fig 11 if the resource planning and cycle time reduction is efficiently carried on during loop 1 itself then any critical error caused while development is negotiated then and there itself ,which saves time and cost incurred for retesting the product.

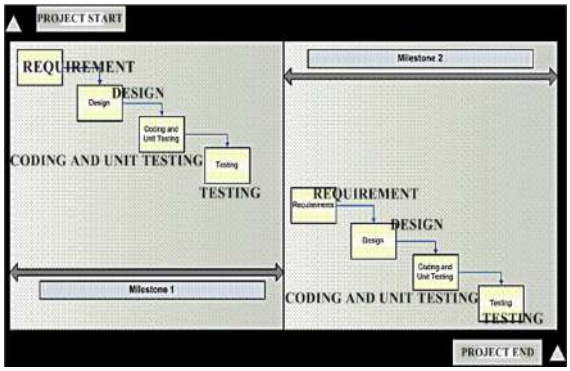

Fig 9: Process start and end

DOI: https://doi-ds.org/doilink/12.2023-34719133

4. DIFFICULTIES FACED DURING MANUAL DEPLOYMENT

- *Available resources may be different*
- *Speed of release is reduced as manually verifying the test cases may take some.*
- *Volume of the testcases constant and verification is bulk work.*
- *Late night releases which causes leads to stay for more than expected time.*
- *Cost for both human resources and release .*
- *Extra computer is required for manual testing.*

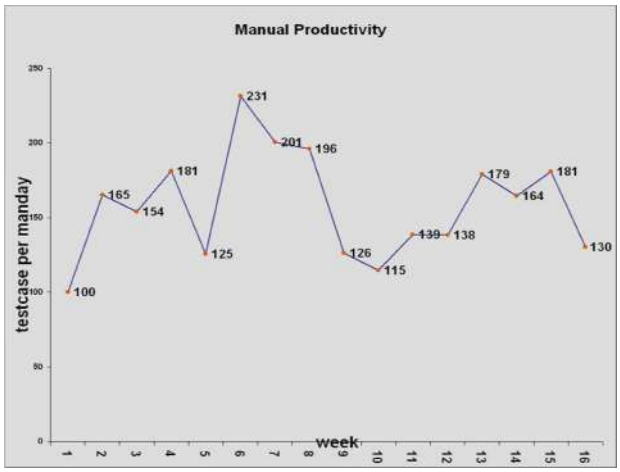

Fig 10: Manual Productivity

5. BENEFITS OF AUTOMATION

- *Extra computer for testing perpose is not required*
- *Automation may increase the speed of releases to down stream users.*
- *Voluminous jobs can be distributed.*
- *Cost saving.*
- *Accuracy is maintained.*
- *Errors caused during manyual testing is reduced Test engineers have their work cutoff.*
- *Stress testing is possible as the test engineers may run the automation tool for several days .*

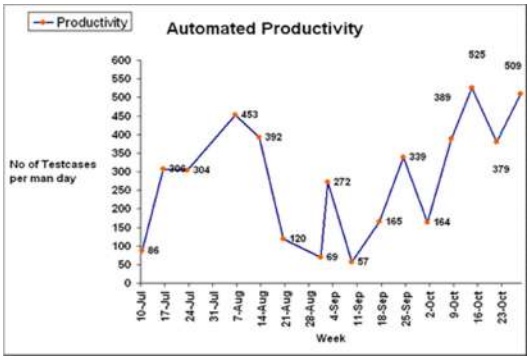

Fig 11: Increase in productivity per due to automation

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

AUTOMATION TOOL

As of now management tools such as minitab, simular, matlab and excel sheets were used to analyze the data but now the real work is to develop an automation tool. Based on the user requirement the automation tool was developed where in excel sheet was the input and basic languages such as C and C# was used to write the source code.

```
// Automated Test System
//System declarations

BEGIN_DECLARATION
 DECLARE_DESCRIPTION("Sanity Automation")  DECLARE_DEVICES(2)
END_DECLARATION
#define Start 13
#define End 384  void TestMain()
{wchar_t
*ModelName,*SWVersion,*checkBox,*ExlValue,*ExlValueToken,*Phon  eValue;
 int row,deviceNumber,ret,i;
// wchar_t *date=__DATE__;
 ExcelWorkbook *wb = DEV1.ExcelOpen(SanitySheetPath,L"Sanity");  wb-
 >SetActiveWorksheet(L"Sanity");
 /* To get the Binary information and the model name*/  ModelName=DEV1.ModelName;
 SWVersion=DEV1.SWVersion;
 wb->SetCellValue(L"F6", SWVersion);  wb->SetCellValue(L"B6", ModelName);
// wb->SetCellValue(L"B7",date);  DEV1.KeyPressEx(L"EEE",10,1000);
 DEV2.KeyPressEx(L"EEE",10,1000);

 PhoneNumber1=DEV1.GettingPhoneNumber(wb,4,'L');
 PhoneNumber2=DEV2.GettingPhoneNumber(wb,5,'L');

 checkBox=(wchar_t*) malloc (sizeof(wchar_t*)*25);  RegionValue=
 (int**)malloc(sizeof(int**)*30);  MenuItem= (wchar_t**)malloc(sizeof(wchar_t**)*30);

 memset(checkBox, NULL, sizeof(wchar_t*)*25);  memset(MenuItem, NULL,
 sizeof(wchar_t**)*25);  memset(RegionValue, NULL, sizeof(wchar_t**)*25);

 for(row=Start;row<=End;row++)
 {
  wb->SetActiveWorksheet(L"Sanity");  checkBox=DEV1.ReadingExcelItem(wb,Select,row);
 if(!wcscmp(checkBox,  L"True"))
 {
```


DOI: https://doi-ds.org/doilink/12.2023-34719133

```
/*Precondtion */
wb->SetActiveWorksheet(L"Precondition");
MenuItem=DEV1.ReadingExcelArray(wb,row,PreDevice1);
if(wcscmp(MenuItem[0],L"Blank")!=0)  DEV1.Condition(wb,row,PreDevice1);

wb->SetActiveWorksheet(L"Precondition");
MenuItem=DEV1.ReadingExcelArray(wb,row,PreDevice2);
if(wcscmp(MenuItem[0],L"Blank")!=0)  DEV2.Condition(wb,row,PreDevice2);

wb->SetActiveWorksheet(L"Precondition");
/* MenuItem=DEV1.ReadingExcelArray(wb,row,PreDevice3)  if(!wcscmp(MenuItem[0],L"Blank"))
DEV3.Condition(wb,row,PreDevice3); */

/*Sanity Part*/
wb->SetActiveWorksheet(L"Sanity");  MenuItem=DEV1.ReadingExcelArray(wb,row,Navigation1);
if(wcscmp(MenuItem[0],L"Blank")!=0)
{
 ItemCount=DEV1.wcharToInt(MenuItem[0]);  DEV1.ExecutingArrayItems(wb,MenuItem,ItemCount);
}
wb->SetActiveWorksheet(L"Sanity");  MenuItem=DEV1.ReadingExcelArray(wb,row,Navigation2);
if(wcscmp(MenuItem[0],L"Blank")!=0)
{
 ItemCount=DEV2.wcharToInt(MenuItem[0]);  DEV2.ExecutingArrayItems(wb,MenuItem,ItemCount);
}
wb->SetActiveWorksheet(L"Sanity");
/* MenuItem=DEV1.ReadingExcelArray(wb,row,Navigation3)
if(!wcscmp(MenuItem[0],L"Blank"))
{
 ItemCount=DEV3.wcharToInt(MenuItem[0]);  DEV3.ExecutingArrayItems(wb,MenuItem,ItemCount);
}*/

wb->SetActiveWorksheet(L"Sanity");

/*Reading Device Number*/  ExlValue=DEV1.ReadingExcelItem(wb,DeviceResult,row);
deviceNumber=DEV1.wcharToInt(ExlValue);

/*Reading Token*/  ExlValueToken=DEV1.ReadingExcelItem(wb,Token,row);

/*Reading Region*/  ExlValue=DEV1.ReadingExcelItem(wb,TokenRegion,row);
RegionValue=DEV1.Parsing(ExlValue);
```

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

```
 if (deviceNumber==1)  PhoneValue=DEV1.PhoneTextRead(ImagePath,RegionValue,127);  else if
 (deviceNumber==2)  PhoneValue=DEV2.PhoneTextRead(ImagePath,RegionValue,127);
/* else if (deviceNumber==3)  PhoneValue=DEV3.PhoneTextRead(wb,RegionValue,127);*/

 sleep(2);  result=DEV1.MakingResult(ExlValueToken,PhoneValue);  if(result==1)
 DEV1.WriteResultInExcel(wb,row,Result,L"Pass");
 else  DEV1.WriteResultInExcel(wb,row,Result,L"Fail");

/*Pasting Image in Excel*/  ret=DEV1.PasteImageInExcel(wb,ImagePath,row,Image);  if(ret ==
 AT_SUCCESS)
  printf("\nPasting image success");  else
  printf("\nPasting image failure");  sleep(5);

 /*PostCondition*/
 wb->SetActiveWorksheet(L"Precondition");  MenuItem=DEV1.ReadingExcelArray(wb,row,PostDevice1);
 if(wcscmp(MenuItem[0],L"Blank")!=0)
   DEV1.Condition(wb,row,PostDevice1);

  wb->SetActiveWorksheet(L"Precondition");  MenuItem=DEV1.ReadingExcelArray(wb,row,PostDevice2);
  if(wcscmp(MenuItem[0],L"Blank")!=0)  DEV2.Condition(wb,row,PostDevice2);

  wb->SetActiveWorksheet(L"Precondition");
 /* MenuItem=DEV1.ReadingExcelArray(wb,row,PostDevice3)  if(!wcscmp(MenuItem[0],L"Blank"))
  DEV3.Condition(wb,row,PostDevice3);*/
 }
}
/* Free the alloacted Pointers*/  for (i=0;i<4;i++)
{
 if ( RegionValue[i] )
  free(RegionValue[i]);
}
for (i=0;i<100;i++)
{
 if ( MenuItem[i] )  free(MenuItem[i]);
}
free(checkBox);  wb->Save();
wb->Close();
```


DOI: https://doi-ds.org/doilink/12.2023-34719133

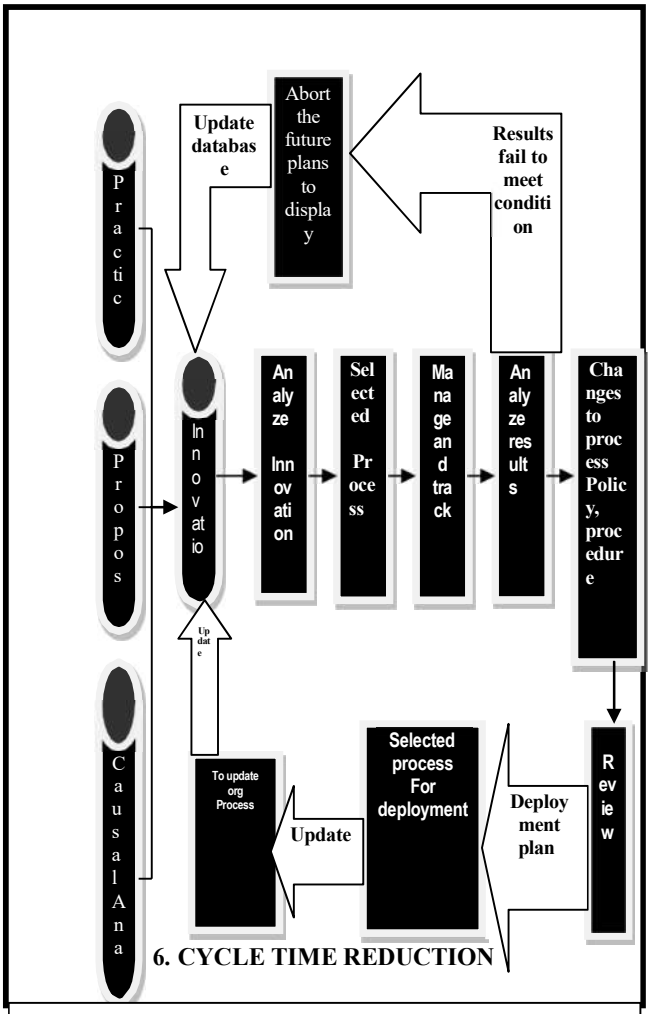

Fig 12: Deployment of automation

Cycle time is the time required to complete a given process. The cycle time required to process a customer order might start with the customer phone call and end with the order being shipped in this example. The overall process is made up of many sub-processes such as order entry, assembly, inspection, packaging, and shipping. The cumulative cycle time of all of the sub-processes in your operation determines when you can promise product to your customer.

Benefits of cycle time reduction?

- Reduced costs
- Streamlined processes
- Improved communications
- Schedule integrity
- Improved on-time delivery

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

- Improved productivity

How can Automation tool help implement a Cycle Time Reduction plan?

Automation tool works with you and your team to understand your business and its processes. Together, we map out your company processes, investigating new methods and identifying opportunities for reducing non- value added activity. Once this is complete, the field engineer helps you create and implement a plan for cycle time reduction.

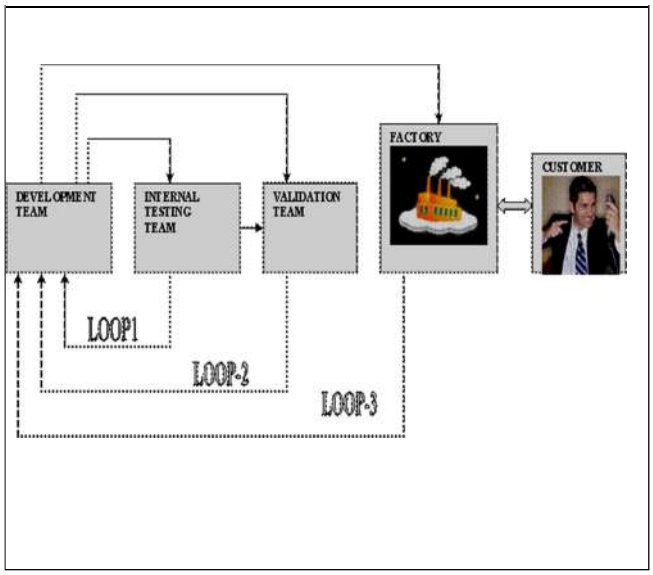

Fig 13: Cycle time reduction.

As shown in Fig.15 if the defects are verified well in advance in loop 1 itself the errors carried on to further processes is reduced.The development team leads and also the test leads are intimated with the errors.so that the leads can take up control measures to overcome errors.

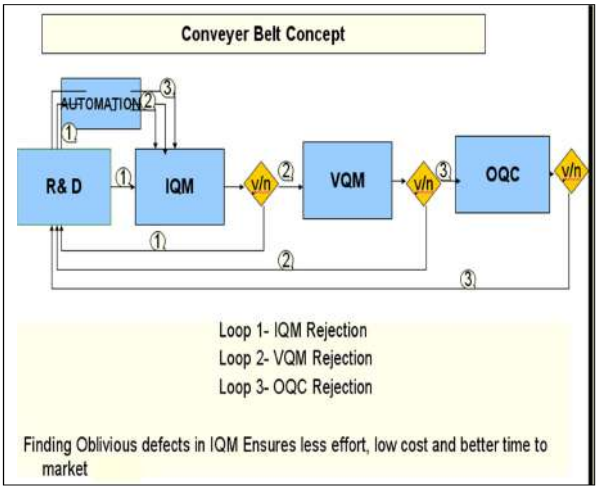

Fig 14: Conveyer belt concept

ISSN: 2456-4265

DOI: https://doi-ds.org/doilink/12.2023-34719133

The conver belt concept is an author specific name given to explain the loops which depect a conveyor belt.As to explain the effect of rejection in an organization ,the cycle time increases if the loop takes a longer path So automation is applied to reduce the cycle time which is easily understood in the Fig .15.

7. CONC LUSION

In this competitive world customer satisfaction is main concern for any company. Hence delivery of the product right in time with quality of the product also maintained certainly takes some pain. As we know that ―out of sight, out of mind‖, the errors carried on to factory and then to distributor, finally to customer leave a bad impact on the reputation of the company. In the FIG.15 if the resource planning and cycle time reduction is efficiently carried on during loop 1 itself then any critical error caused while development leads are negotiated then and there itself ,which saves time and cost incurred for retesting the product. An automation tool can help human resource and extra computer required for testing, voluminous jobs can be distributed, speed of testing is increased and finally overall cost is reduced .The second issue which helps in scheduling the release to downstream users which causes server down scenario which can be reduced, the test leads as well as the development leads will know when to release the data , also to predict and optimize overall cycle time of server based on understanding of upload sequencing ,server load and analysis of bottle necks in network. To meet the current load a new server has to be installed which is costly, optimization of cycle time will handle this issue efficiently.

Parameter	Manual Execution (What IF Analysis)	Automation
No of Cases	94295 test cases	94295 test cases
Test efficiency	157.94	283.12
No of engineer	4	4
Man month	7months	4month
Effort required	27 man month	15 man month
cost	27000	15000
Saving		**12000 USD**

Figure 15: Comparison of manual and automation result

DOI: https://doi-ds.org/doilink/12.2023-34719133

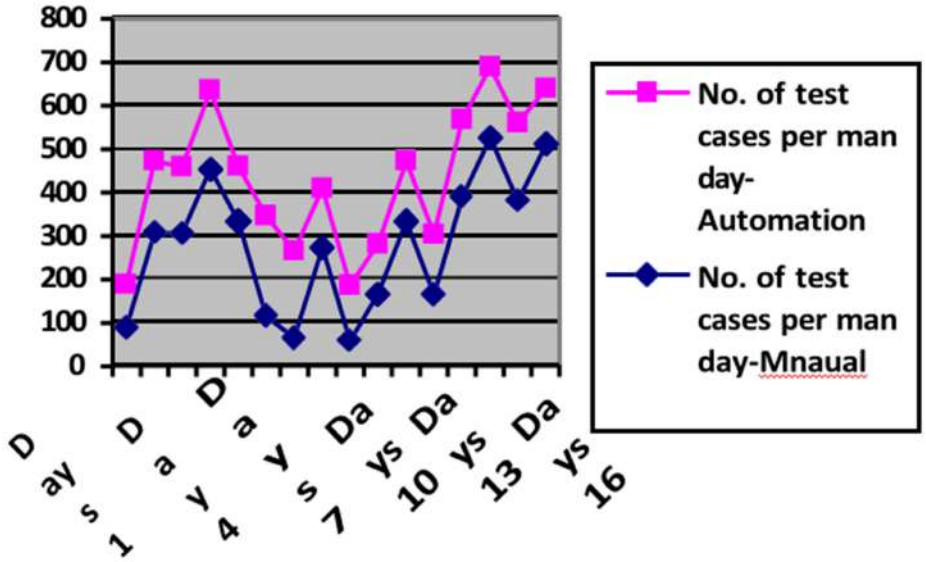

Figure 16: Increase in productivity due to automation

REFERENCES

[1] Dr. Bhargav Gangadhara, "Optimizing Cloud - Based Manufacturing: A Study on Service and Development Models", International Journal of Science and Research (IJSR), Volume 12 Issue 6, June 2023, pp. 2487-2491,https://www.ijsr.net/getabstract.php?paperid=SR23626155823

[2] Dr.Bhargav Gangadhara , "Optimize Utility in Computing-Based Manufacturing Systems Using Service Models and Development Models", Presented and published at International Journal of Computer Science and Information Security, IJCSIS Editorial Board, Vol -14, Page 5.

[3] Dr.Bhargav Gangadhara,"Optimize utility in cloud-Based manufacturing systems using service models and development models", Presented and published at International Journal of Innovative Research in Advanced Engineering.IJIRAE Editorial Board, August 11 - 12, 2016.