Security Tickets Monitoring System

Sameeksha Shetty[1], Smriti Srivastava[2]

[1]*Student, Department of Computer Science, Rashtreeya Vidyalaya College of Engineering(RVCE), Karnataka, India.*
[2]*Assistant Professor, Department of Computer Science, Rashtreeya Vidyalaya College of Engineering(RVCE), Karnataka, India.*

---***---

Abstract - *Agile Project Management is a revolutionary technique used to better the overseeing process of the creation of applications. In this case mainly the development, deployment and maintenance of the applications. This style of management, coupled with the Jira Project Management Software makes the process greatly more effective in monitoring and aiding the applications' progress. Even so, the type of manual work undertaken by the managing authorities is considerably taxing. With the development of an automation system to take over most of the redundant work, the management process will be considerably easier. The tasks that will be taken over by the system are lengthy, monotonous and repetitive like the periodic report generation process. The report required all the data on these applications to be compiled and processed regularly in a similar manner. Such work would be taken over by the automation system to free up more time and energy to be directed to other aspects of the managerial process that demand it.*

***Key Words*: Automation, Project management, Agile project management, Jira software, REST API, Java, Excel.**

1. INTRODUCTION

The process of project management is a higher level view of the ongoing of planning, developing, deploying and maintenance phases of the applications creation. The act of management plays a huge role in how the applications in consideration turn out in terms of progress and quality. This bigger picture that the managers get to see and analyze helps them adjust or calibrate the progress throughout the lifecycle of the application, thus making sure that the direction that the application is taking is in line with the vision that it was intended for. The managers analyze the details relevant to this work without extensive documentation and therefore many tools and techniques have been introduced to aid the job.

The technique in this case is agile project management. Agile project management is a type of project management style that has become very popular recently. It has been vouched for as a technology that speeds up the entire process in comparison to the traditional approach. It even lowers the cost. These results are brought forth mainly by the four values specified in the Agile Manifesto. The values include:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

The four values are used differently in different Agile methodologies such as:

- Extreme programming
- Scrum
- Kanban etc.

Even though the methodology provides such improvement to the earlier ways of carrying out the process, the managers are required to comprehend the state of the application and its progress without descriptive documents and therefore, various tools to aid the Agile project management process popped up.

The software that will be focused on in this paper is Jira Software. Jira is a popular project management tool used by software professionals. Jira is particularly an agile project management supporting different agile methodologies like scrum, kanban, a combination. This tool allows for the creation of dashboards that help track and manage an amount of applications. They give us an extensive view of the incoming issues, tickets, requests etc. being raised by teams throughout the course working on certain applications. The collective picture of all the tickets from all applications is viewed by a manager in charge of those applications from their Jira dashboard. This information is taken and used to create a report on the current status of the application and an analysis of that data. This report generation practice is carried out frequently and therefore poses as a great use of company resources. It not only uses valuable time and effort but is also periodically repeated, making the act even more redundant. This is the main process that would benefit from automation.

The model that is described in this paper has the main goal of automating the report generation phase of management and delivering analysed data for the tickets in the report. The model will automate the generation of the reports on a defined periodic basis along with distribution.

2. LITERATURE SURVEY

The papers referred to have helped to come up with a more whole system that performs the necessary processes required to complete the objectives.

The papers surveyed were:

The paper by Deniz Özkan and Alok Mishra takes a look of the Agile methodology and its advantages that have made its use widely popular. They do touch on the difficulties faced by managers that are using this approach and lists useful tools to aid the process. The objective of this paper is to analyze the agile development methodologies and management approach used in developing a complex software project. This further demonstrates how to overcome risks and barriers in each development phase of such complex inventive software projects. It also provides a set of guidelines regarding how the agile methodologies can be adopted, combined and used in these kinds of complex software projects.

The paper written by Mishra D.and A. Mishra analyses the Agile approach with respect to complex applications and explores how one can overcome problems that are usually encountered in each of its phases.

Dr. B.E. Narkhede and Dr. M.M. Chandane spell out the various complications with agile project management and how automation can be used to deal with each of them.

Andy Neumann, Nuno Laranjeiro and Jorge Bernardino talk about Rest APIs in their paper. They mention the ways in which web services are extensively used and to facilitate their use, we may employ Rest APIs They analyse a ton of REST (Representational State Transfer) APIs and describe the best practices that can be adopted.

Arpitha Prasad and Nirmala H. describe the use of Rest APIs specifically provided by Atlassian for interaction with the Jira software. The process of connecting to a server and asking for data is described as well as syntaxes provided in the language Curl.

From these various sources, a clearer understanding of the methodology to be adopted can be formed.

3. PROPOSED SYSTEM

3.1 System Architecture Overview

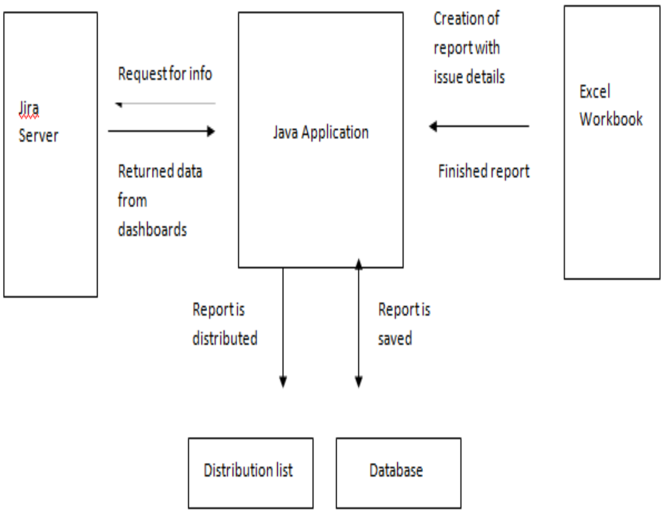

Figure -1: Overview of system architecture

The figure-1 depicts the three main modules of the system along with an overview of the interactions taking place between them.

4. APPROACH TO BUILDING THE SYSTEM

Automation being the main focus of the project, its development is centered on this aspect. Our system performs certain operations for the main target of generating and distributing a report and all of these phases must be auto mated. The triggering of the report generation is to be automated as well. Therefore to achieve this, each stage uses certain methods to perform its activities.

Firstly, the data acquisition phase is to be programmatically carried out by the use of the Jira REST API. It provides us a way of securely connecting to the Jira server and getting data in the form of ticket attributes in return. The requests for this data is sent through REST functions defined by Atlassian and we use the Jirs Rest Client instance created on establishing an authentic connection to carry out these transactions.

The data received is stored and computations are performed on it mainly with respect to time and other basic calculations. The time calculations are to be carried out by using a java package 'java.time'.

After this, the data is to be exported to and formatted in an excel workbook with classifications and ordering of the data. These operations are also programmatically automated with the help of apache POI functions.

5. METHODOLOGY

The proposed system is carried out in the following phases:

1) Optimizing the Jira Dashboard:

The data coming in from all applications under surveillance can be found on the dashboard available to the manager for observation and analysis. These dash boards are organized to produce a better view of the tickets in real time as well as creating statistical representations of the current tickets that have arrived. This step works as a preprocessing of the data to be used in the project.

2) Establishing a connection with the Jira Server:

In a java project with maven framework and the necessary Jira and Apache POI dependencies added in the pom.xml file, we establish a connection with the server programmatically with the use of Jira's REST API methods. We send the login credentials to the server along with the URI of the Jira login webpage. With basic Http authentication, the server establishes a secure line of communication in the form of a Jira rest client instance that is returned to us for further use.

3) Retrieving data from the dashboards:

We use the created client to get a search client result using a JQL statement. JQL statements are Jira Query Language statements meant in particular for the function of retrieving tickets from Jira. We store the returned issues in a Promise set and repeat the step for all required JQLs. The different applications' tickets have been called separately for ease of report creation.

4) Using Apache POI to create a new report:

Using the Apache POI functions a new Excel file and workbook is created as well as different cell formatting styles for overdue alert creation at a later on stage. The three types are green, light red and red which represent the ticket in different phases of their ageing process since creation to or beyond production due date.

5) Placing the data into sheets:

For each JQL query:

We have two user defined functions

a) createObj: The attributes of this function are the jira rest client and the application name. What this function does is it fills up a two dimensional array object with details of the issues. Each row represents each ticket retrieved and will be filled with values of each field present on the Jira dashboard. This representation makes for an easy transition from array object to tables in Excel. The fields accessed are either system defined fields or custom made fields. The system fields can be accessed through system functions directly but hen accessing custom fields we must use a custom field accesser object that take either the field id or name as input and retrieve the field from the server as well as the relevant values for that particular issue. This way we access all field values that we want to evaluate such as app portfolio id, project that the issue is associated with, issue key, assignee of the issue, summary of issue raised, creation date of issue, scanning source of issue, risk rating, labels associated with issue, update date of issue, priority of issue, production due date of issue, status of issue, etc. These fields were ones that were accessed from the dashboard. We also create new values for the ageing of the ticket as well as the overdue alerts. The ageing is considered as the number of days from the creation of the tickets to the present date. The overdue values are the difference between the current date to the production due date.

b) fillTable: This function gets the array of issues and field values as its argument along with the sheet that the data is to be put into. It uses the apache poi functions to transfer the object data into the sheet as tables. The table is formatted as necessary and then we create overdue alerts. The alerts are based on the overdue column values. Three types exist, as discussed. The green is from tickets that are beyond 30 days difference with the production due date. Light red is for fifteen days to thirty days and red is for those within fifteen or those that have already passed the production due date.

The functions are present in the interface extended by the main class. These two functions are called for every JQL passed to the Jira server that returns issues. The issues are retrieved in accordance to the application so that we may place each application's issues in one sheet, thus providing a complete overview of the application's progress in the report. The fillTable function further orders the issues in accordance with their security scan types. For example, the IQ scan tickets will be listed first, followed by the Fortify and so on.

6) Scheduling the execution of the code:

We then program the execution of the code to occur in a periodic manner, say every day or week.

7) A database is created:

To record all the previous executions, we create a database that stores the reports as they are being generated.

8) Deploying the project:

The system is then deployed onto the server that the manager can access. This server can perform the job of distributing the reports directly to the concerned authorities.

6. EVALUATION

From experimental evaluation we can say that the time taken is an average of 10 minutes for a regular dataset and considering the amount of time taken for a manual version of this process, this is a great improvement. Regularly, this process can take about 45 to 60 minutes and the fact that this process id completely automated, including the order to start the process, the automation is effective. The results also show that the reports generated always have a completely accurate representation of the dashboards status.

7. CONCLUSION

It is possible to say that the system serves its intended purpose by saving the manager a lot of manual labor. By collecting and streamlining the dataflow of the tickets right from when they are taken from the dashboard to when they are viewed in the workbook, a large amount of time is saved as well. The objective of creating this project was to aid and speedup the daily or weekly management of the applications and the result has managed to reach this goal by generating reports that will be much easier to analyse and interpret in order to deduce the status of these applications. The objective of job of the manager much easier by taking over redundant work was actualized in this manner so that more time and energy may be freed up for better use in more challenging tasks.

REFERENCES

[1] Deniz Özkan and Alok Mishra ,"Agile Project Management Tools: A Brief Comprative View", Cybernetics and Information Technologies, Turkey, 2019, Vol. 19, No 4

[2] Mishra, D., A. Mishra. "Complex Software Project Development: Agile Methods Adoption" – Journalof Software Maintenanceand Evolution: Researchand Practice, Vol. 23, 2011, No 8, pp. 549-564.

[3] Arpitha Prasad and Nirmala H., "Survey on JIRA INTEGRATION USING REST APIs", Department of Computer Science and Engineering, SJB Institute of Technology, Bangalore, India, 2016

[4] Dr. B.E. Narkhede2, Dr. M.M. Chandane, Journal of Business and Management, Volume 17, Issue 8.Ver. I, Aug. 2015, Sachin D. Pandkar1

[5] Andy Neumann, Nuno Laranjeiro, Jorge Bernardino, "An Analysis of Public REST Web Service APIs", IEEE Transactions on Services Computing , June 2018

[6] Rahul Shende (2012) , "Software Automation Testing Tools for Beginners"

[7] Agile Menifesto Beck, Kent; et al. (2001)."Manifesto for Agile Software Development". Agile Alliance

[8] Scott Ocamb (2013, November7). Agile Project Management Principles9. Jan Juricek, "Agile Project Management Principles", Lecture Notes on Software Engineering, Vol. 2, No. 2, May 2014

[9] Dubakov, M.P. Stevens. "Agile Tools: The Good, the Bad and the Ugly." Report, Target Process, Inc, 2008.

[10] Taheri, M.S. M.Sadjadi. "A Feature-Based Tool-Selection Classification for Agile Software Development." – In: SEKE, 20