International Journal of Interdisciplinary Innovative Research &Development (IJIIRD)
ISSN: 2456-236X
Vol. 05 Issue 02 | 2021

CI/CD Deployment: Automating CI/CD Deployment Using RASA Framework

[1]Mr. Varun N Dhanpal, [2]Vignesh S
[1]*MCA Scholar, School of CS & IT, Dept of MCA, Jain (Deemed-to-be) University, Bengaluru*
[2]*Asst Professor, School of CS & IT, Dept of MCA, Jain (Deemed-to-be) University, Bengaluru*

ABSTRACT

CI/CD was first introduced in 1991 it's a practice for merging all the working code of all the developers into a centralized repository. In CI/CD the CI stands for continuous integration and CD stands for continuous deployment or continuous delivery. According to continuous delivery any changes made to the code developed by a user is automatically deployed to a specified repository after it's tested for bugs. The main reason for doing so is to ensure that there is continuous delivery of the deployed code and to ensure that there is minimal downtime in deployment of the new code. According to continuous deployment instead of the code going to the repository it instead goes from the repository to deployment. This ensures that the deployment team doesn't have worry about the abundance of manual operations.

Keyword: CI/CD Deployment, Continuous Deployment, Continuous Integration and Continuous Delivery.

1. INTRODUCTION

Even though the CI/CD pipeline is created to ensure that continuous deployment there is still a lot of manual work involved. This project aims to reduce the manual work for the developers by automating the process of developing. In the current process each developer has to manually go and enter the main three parameters for deployment i.e., Jira Ticket ID, Environment and the Repository URL link. These three parameters are mainly needed for deployment in the specified environment. Usually, the user has to go to the deployment portal and enter these parameters manually. The main aim of this project is to integrate a chat bot with slack so that the user can provide the parameter details to the chat bot. The rest of the processes would then be handled by a backend orchestrator.

This is done in order to ensure that the user doesn't have to manually go to the deployment page and enter the details. Instead, the user can have a conversation with the chat bot and deploy much faster. This also allows for multiple deployments to take place at the same time. This is possible by the thread feature of slack where multiple conversations can take place in the same window. The overall architecture of the project looks similar to this.

2. LITERATURE REVIEW

[1] In this paper the author talks about how continuous integration & continuous deployment enable organizations to release feature enriched products at a faster rate. It also conducts a thorough research on all the best practices and challenges faced in the art of continuous practices such as continuous deployment and integration. It compares various pre published papers and outlines all the challenges, practices and gaps for future research in the same field.

[2] The author of this paper aims to improve the robustness and effectiveness of the present CI/CD pipeline solutions. The paper talks about the various methodologies and technical solutions on how the CI/CD pipeline can be improved. The paper also points out the common mistakes made in the initial configuration of the pipeline itself. The paper also highlights the multiple uses of the CI/CD pipeline and also shows how we can overcome these common limitations by using some basic best practices in our configuration of the pipeline.

[3] In this paper the authors talk about the widespread uses of bots in the software industry. It talks about how software bots are now being used for multiple tasks. It explains how bots are preferred in some places compared to their human counterparts. This is mainly due to the bots everlasting enthusiastic persona. It also talks about how bot interactions have improved over time to such an extent that now bot makers have to ensure that the user knows that they are chatting with a bot.

3. PROBLEM STATEMENT

Even though the CI/CD pipeline aims at decreasing the amount of manual work that is required for deployment of the various versions of code, there are still a lot of manual tasks that have to be taken care of. The main aim of the bot is to reduce these manual tasks and facilitate multiple deployments simultaneously.

4. PROPOSED SYSTEM

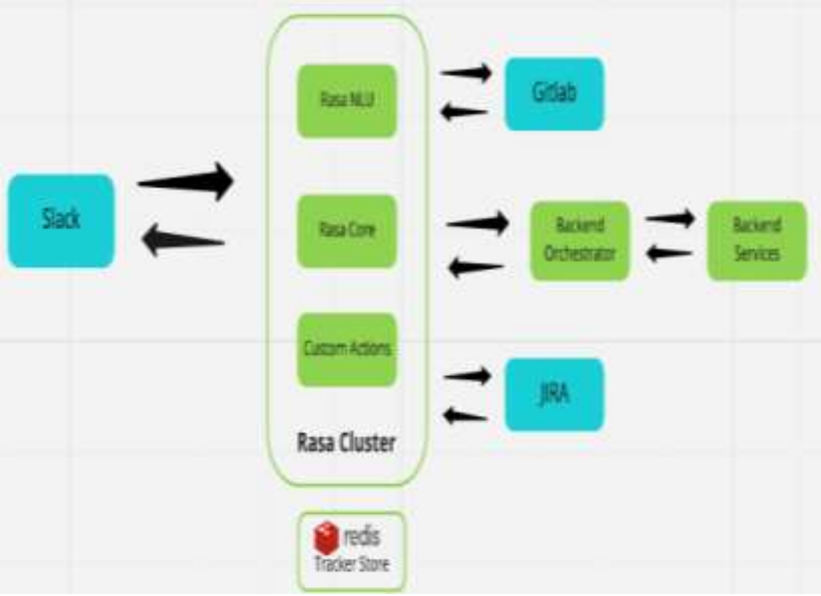

Figure 1 - Architecture Diagram

In the proposed architecture the main part is played by the RASA framework. It provides the chat bot with its persona and the conversation capabilities (RASA NLU). In the RASA framework we can mention all the possible conversations and the responsive actions as well. The RASA framework then communicates with a backend orchestrator for facilitating the deployment. The orchestrator uses multiple API's to perform various checks on all the provided input parameters and to deploy to the specified environment. The overall front-end/client end of the bot is through slack. The user can enter into a conversation with the bot on slack like you would with any other normal user.

5. METHODOLOGIES

There are various applications that are being used in conjunction to help with the automation of the deployment.

- [4]Slack – Slack is business oriented messaging service developed by slack enterprise situated in America. Slack mainly acts as a messaging service but also allows for integration with various other API's as well. This allows the user to build apps within slack that can assist the user. Such as chat bots.
- [5]RASA Framework – RASA is an open source framework created by the team at RASA. They provide a standard infrastructure layer to create a conversational AI.

Here are some important features and parts of the RASA framework.

- Action – A step taken by the [6]bot in the conversation taking place, Such as a response to the user. Even actions such as calling an API are included in this.
- Entity – It refers to the bots ability to extract certain keywords from the users' conversation. This could refer to details like mobile numbers, address etc.
- Form – It is a custom action provided by RASA, with the user of which the bot can request multiple user inputs in a sequential order.
- Intent – [7]It refers to the bots ability to understand what the user is trying to convey or accomplish with the given message. For example the bot needs to understand whether the bot is greeting or asking a question.
- RASA Core – The open source dialog engine provided by RASA. It's the engine that decides where to take the conversation next depending on the specified context.
- RASA NLU – It is the Natural Language Understanding part of the RASA framework. Its main role is to classify intents and extract the required entities from the user messages.
- As of RASA 1.x both RASA Core and RASA NLU are included together in a single package known as NLU component.
- [8]Swagger UI – It is fully hosted in the swagger hub. It provides a fully interactive visual AI with which the developers can interact. Developers can interact with API's without having any pre implemented logic in place.

6. CONCLUSION

With all the mentioned components and technologies we can create a more streamlined way of deployment in the CI/CD pipeline. With this we can facilitate multiple deployments simultaneously and also decrease the time taken for deployment significantly as lesser manual work is required by the user to do the same tasks.

7. REFERENCES

[1] Mojtaba and Ali Babar, Muhammad and Zhu, Liming Shahin, "ontinuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices," IEEE Access, vol. 5, no. 10.1109/ACCESS.2017.2685629, pp. 3909-3943, 2017.
[2] Keheliya Gallaba, "Improving the Robustness and Efficiency of Continuous Integration and Deployment," 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), no. 10.1109/ICSME.2019.00099, pp. 619-623, 2019.
[3] Carlene and Storey, Margaret-Anne and Zagalsky, Alexey Lebeuf, "Software Bots," IEEE Software, vol. 35 , no. Issue 1 - 10.1109/MS.2017.4541027, pp. 18-23, 2018.
[4] Slack Enterprise. Slack. [Online]. https://slack.com/intl/en-in/
[5] Rasa. (2021, May) Rasa Docs. [Online]. https://rasa.com/docs/rasa/
[6] Menal Dahiya, "A Tool of Conversation: Chatbot," INTERNATIONAL JOURNAL OF COMPUTER SCIENCES AND ENGINEERING, vol. 5, pp. 158-161, May 2017.
[7] Dr. John Woods Sameera A. Abdul-Kader, "Survey on Chatbot Design Techniques in Speech," (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 6, no. 7, pp. 72-80, 2015.
[8] Swagger. Swagger UI. [Online]. https://swagger.io/tools/swagger-ui/