Vitality Web Services

Sanidhya Jain , Shubham Verma , Somya Mehta , Suyash Pathak , Vaibhav Rathore , Vinay Bagul
Computer Science and Engineering, Acropolis Institute of Technology and Research, Indore, India

Abstract: - The purpose of Vitality Web Services is to automate the container-based cloud Platforms as a Service (PaaS). Through this application, developers can deploy, manage, update and scale up their big-scale projects.

This platform is elegant, flexible, and easy to use, offering developers the most straightforward path to getting their apps to market which is fully managed, giving developers the freedom to focus on their core product without the distraction of maintaining servers, hardware, or infrastructure. We need external controllers that must function within the constraints of the cloud providers' APIs.

Keywords: - Platforms as a Service, cloud

1. Introduction

We outline the various procedures required to install apps utilising such an external controller in this document. The proposed taxonomy is then used to evaluate a number of management tools, including Chef, SaltStack, and Ansible, for application automation on cloud computing services based on the established paradigm.
Since Vitality Web Services(VWS) is fully managed, developers have the freedom to concentrate on their core product without being distracted by infrastructure, servers, or hardware upkeep. External controllers are required, and they must operate within the limitations of the cloud providers' APIs.

2. ProblemFormulation

The target audience for this web application is mainly developers and project leads who are working from different organizations. Our project will cover the gaps the loopholes carried by the manual project management system by developing an integrated system by dodging the use of external applications and software.

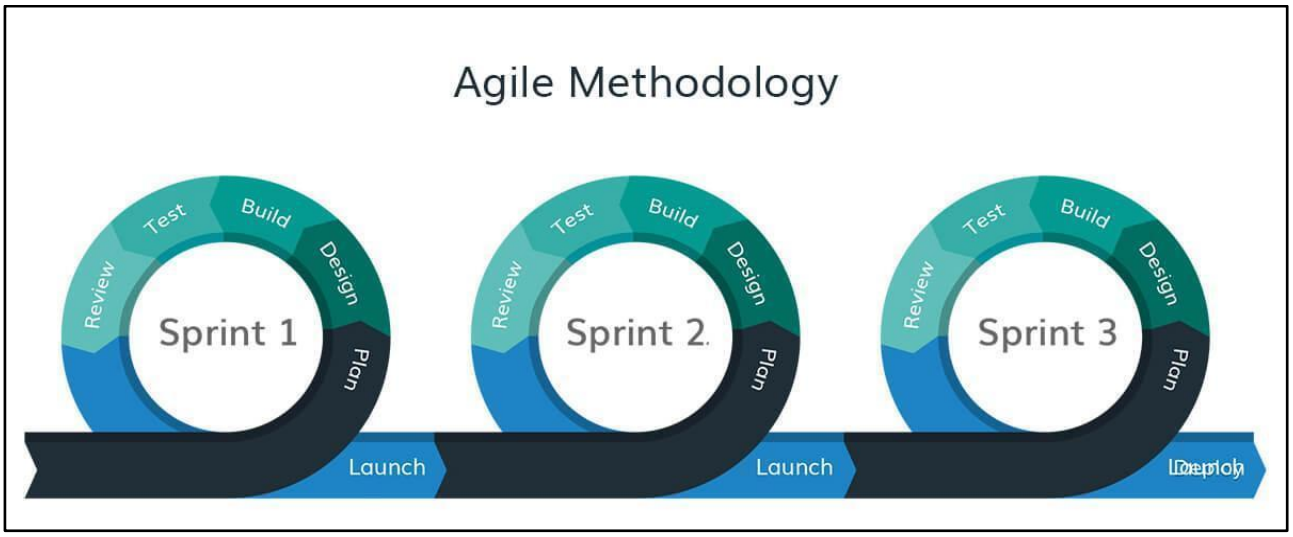
- To assist the staff in capturing the effort spent on their respective working areas.
- Developers are liberated from having to worry about maintaining infrastructure, servers, or hardware and may instead focus on creating their primary product.
- The system generates types of information that can be used for various purposes.
- Vitality Web Services, businesses can focus on creating and deploying apps that start generating revenue right away. Building a platform as a service architecture is the focus of our project.
- Be easy to understand by the user and operator.
- Have a good user interface.
- Delivered on schedule within the budget.

3. Literature Review

Deploying static web pages on a server with file hosting add-ons would be our first major objective.
The second component would be a Linux virtual machine with many scalable parameters that supported several different languages, such as node.js, python, java, SpringBoot etc. enabling the creation of custom server-side applications by users.
The quickest path to becoming an app firm for a business is through Vitality Web Services. This is a solution that lets businesses devote their attention to creating and deploying apps that start generating revenue right away. Building a platform as a service architecture is the focus of our project.

4. Methodology

The methodology of the agile model is followed in this project.
The Agile software development methodology is one of the simplest and most effective processes to turn a vision for a business need into software solutions. Agile is a term used to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change.



The various phases of the Agile model are as follows:

- Requirements
- Design
- Development and Coding
- Integration and Testing
- Implementation and Deployment
- Review

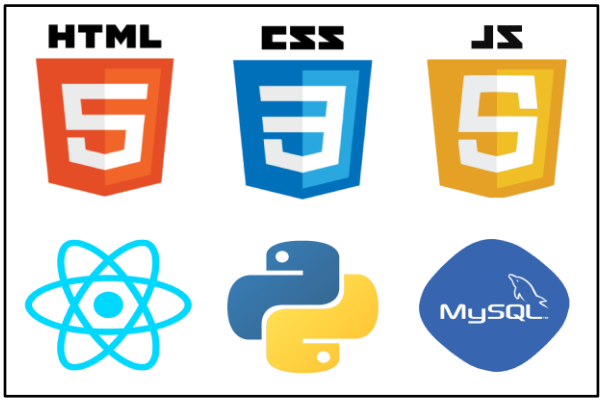
The project will focus on various functionalities and work on the development of the same. The development is divided into multiple releases which will focus on different features, and then all the features will be combined, resulting in one single application. The requirements will be modified according to the resources and needs.

4.1 Technology Used:

(a) Front-end:
- HTML5
- CSS
- Javascript
- React

(b) Back-end:
- Python(Django framework)

(c) Database:
- MySQL



5. Result Discussions

The project releases cloud Platforms as a Service based on containers (PaaS). This web application is used by developers to scale, manage, and deploy contemporary apps. Our platform is beautiful, adaptable, and simple to use, giving developers the simplest way to release their apps.

6. Conclusion

The programme is a cloud Platform as a Service built on containers (PaaS). This software is used by developers to scale, manage, and deploy contemporary apps. Our platform is beautiful, adaptable, and simple to use, giving developers the simplest way to release their apps. There has never been a more pressing demand for the efficiency and agility that the cloud provides. Those companies that fail to rapidly expand their use of the cloud and exploit these benefits risk falling even further behind those who do.

Acknowledgment

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have gotten this all along with the completion of my project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

We respect and thank **Prof. Shivshankar Rajput**, for providing me with an opportunity to do the project work at **Acropolis Institute of Technology and Research** and for giving us all support and guidance which made me complete the project duly. We are extremely thankful to him for providing such nice support and guidance, although he had a busy schedule managing corporate affairs.

We owe my deep gratitude to our project guide **Prof. Shivshankar Rajput**, who took a keen interest in our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

References

[1] **https://www.python.org/**

[2] https://reactjs.org/

[3] **https://developer.mozilla.org/en-US/docs/Web/JavaScript**

[4] https://www.djangoproject.com/

[5] **https://www.mysql.com/**