Shahjalal University Of Science And Technology, Sylhet

Software Engineering, IICT

SWE-420

Internship Report

Submitted By:
Md. Mehedi Hasan
Registration No: 2017831023
Department of SWE
IICT, SUST

Performed At:
Vivasoft Limited
Floor 16, Ahmed Tower, Dhaka

Time Period: 09/21 - 03/22

Date of Submission: July 16, 2022

July 16, 2022

LETTER OF TRANSMITTAL

July 16, 2022

The Director

Institute of Information and Communication Technology

Shahjalal University of Science and Technology

Dear Sir,

I would like to submit my internship report as a part of internship program. Giving the opportunity to relate my academic knowledge with recent software trends in renowned software industry and make the opportunity to gather great experience from industry, I am so thankful to the Department of Software Engineering, IICT, SUST. This report is based on my learning and experience during my internship time period, starting from September 1, 2021 to March 30, 2022. This report both covers the technical skills that I gained through my internship program and represents my project involvements, acquired skills and other improvements and issues. It is believed that the report will exhibit the overall outcomes of my internship course. I am looking forward to your effective feedbacks on my entire report.

Sincerely yours,

Md. Mehedi Hasan

Registration No: 2017831023

Department of Software Engineering

IICT, SUST

Letter of Endorsement

To Whom It May Concern

Subject: Approval of the Report

This is to certify that the report entitled with **"Internship Report"** has been submitted to me prior to final submission.

I have personally gone through and reviewed all the contents of the report and did not find anything violating the company's privacy rules and regulations. Also, I confirm that the information provided here are authentic and not exaggerated.

I, therefore gladly state the validity of the report and wish *Md. Mehedi Hasan* a bright future.

Shafiul Hasan

Shafiul Hasan Md Tareq
Managing Director
Vivasoft Limited

ACKNOWLEDGEMENT

I am very much delighted to present this report about my internship experience at **Vivasoft Limited**. I offer my gratitude to the Department of Software Engineering, SUST to arrange an opportunity of my internship at Vivasoft Limited. I also express my regards to Vivasoft Limited.

From Vivasoft, I would like to thank **Mr. Shafiul Hasan Tareq**, Founder of Vivasoft Limited, who supported me throughout the internship period. I would also like to thank my mentors, **S.M Ariful Islam Tusher**(Software Engineer) and **Moazzam Hossan Rasel**(Lead Devops) who trained and guide me as well as made me familiar with the culture and customs of Vivasoft Limited.

At last, I would like to thank every member of Vivasoft and my intern-mates who showed me such kind and generous gesture throughout my internship period.

EXECUTIVE SUMMARY

This internship report serves the purpose to record the details of my industrial training which was conduct in Vivasoft Limited, which is one of the audit software company in Bangladesh. This report will cover the details of my internship in the software engineering field for a duration of six months which began from 1 September 2021 till 30 March 2022 at Vivasoft Limited which is located at Floor 16, Ahmed Tower, 28, 30, 1213 Kamal Ataturk Ave, Dhaka 1213.

Students who are undertaking the Software Engineering in IICT, SUST are compulsory to undergo an industrial training for a period of six months prior to graduating. Students are allowed to make their own choices to enter any audit firms or company to do their internship, as long as the audit firm is recognized. This is to ensure that the students will be getting the exposure to involving in industrial environment. The objective of having this industrial training are to develop a deeper understanding on the course which they are undertaking and to provide the appropriate work-related training to them in the field of software engineering profession.

Contents

1 Introduction **7**

1.1 General outline of internship 8

1.2 Objectives 8

1.3 Literature Review 8

1.4 Source of Information 9

1.5 Scopes 9

1.6 Limitations 9

2 Company Profile **11**

2.1 Company overview 11

2.2 Management 13

2.3 Location and Physical Layout 13

2.4 What Vivasoft Does 14

2.5 Vivasoft Products 15

2.6 Tools and Frameworks 18

2.7 Domestic Environment 20

2.8 Lunch and snacks 20

2.9 Recreation 20

3 Internship Activites **24**

3.1 Things I learned at Vivasoft 24

3.1.1 Golang 24

3.1.2 Containerization 25

3.1.3 Docker 26

3.2 Kubernetes 29

3.3 Docker Swarm . . . 31
3.4 Helm . . . 32
3.5 Terraform . . . 33
3.6 Tsung . . . 34
3.7 Jenkins . . . 35
3.8 Amazon Web Service . . . 37
3.8.1 Route53 . . . 38
3.8.2 EC2 . . . 38
3.8.3 VPC . . . 39
3.8.4 RDS . . . 39
3.8.5 DynamoDB . . . 40
3.8.6 S3 . . . 41
3.8.7 CloudFront . . . 41
3.8.8 IAM . . . 42
3.8.9 ACM . . . 43
3.8.10 Codepipeline . . . 43
3.8.11 ECS . . . 43
3.8.12 EKS . . . 43
3.8.13 ECR . . . 44
3.9 Non Technical Aspects . . . 44
3.10 Things I did at Vivasoft . . . 45
3.10.1 GPS Server Deployment . . . 45
3.10.2 Wellteam Deployment . . . 50
3.10.3 Some screen shots . . . 52

4 Conclusion **54**

Chapter 1

Introduction

An internship is an opportunity offered by an employer to qualified employees, called interns, to work at a firm or company for a fixed period of time. It can be paid or voluntary. Interns are usually undergraduates or newly graduated. Internships are usually undergraduates or newly graduated. Internships are usually part-time if offered during a university semester and full-time if offered during the vacation periods.

Interns are trained in organization to be a suitable employee for the industry in the upcoming professional life. In this way, both industry and university can understand the gap between those through the interns. it is an opportunity of the students to gather some professional experience before entering the industry.

Department of Software Engineering, IICT, SUST provides the opportunity to their students to experience a six-month internship as a part of their academic curriculum in the 7th semester. SWE always gives emphasis on both academic studies and industry practices to keep up-to-date with the latest technologies. The internship helps to add a great value to personal qualifications and experiences.

This section is divided into several subsections. In subsection 1.1, the general outline of the internship is discussed. Objectives, literature review, sources of information, scoped and limitations has been discussed in subsections 1.2, 1.3, 1.4, 1.5 and 1.6 respectively.

1.1 General outline of internship

Internship has no fixed outline for every company. It varies from organization to organization. Internship can be generally divided into two types. These are:

1. Part Time
2. Full Time

Part time internship is mainly hour based and suitable when the students are also attending a university. In full time internship, interns are almost similar to a full-time employees as they also spend the full working hour in the office. Full time internship is better for students having a vacation going on or not attending a university at that time. It provides more hands-on experience than part time internship. My internship was on full-time basis.

1.2 Objectives

Internship has also some specific and defined aims. Some key points regarding the objectives of internship are discussed below:

1. Getting familiar with the corporate environment
2. Bridging the gap between industry and academia
3. Idea about current and future industry requirements
4. Gain professional attitude, growth, maturity and judgment
5. Hands on experience

1.3 Literature Review

The report is mainly based on my internship experience at **Vivasoft Limited**. In order to follow a standard literature, firstly detail description regarding the company structure, culture and the way of management are described. Next, there has been discussion on the projects I was involved in and the experiences, skills that I have gathered while working

on the projects. This part contains the technical details about my work here. Lastly, my personal opinion regarding the whole internship period is also inclued.

Please note that, I have only discussed what I am permitted to share from the organization to the outside world

1.4 Source of Information

This report contains much technical and theoreitical information. All the information presented here is authentic. The authenticity is based on by the review of Vivasoft Limited. The external information is based on the related website links. Information related to the company was verified by the required personnel before submission in order to check the flaws regarding the accuracy and confidentiality. The external information included on this report is linked with the reference section for proof of its authenticity. Also, the last access time is provided for alleviating any probable confusion.

1.5 Scopes

Scopes mean the area of focus and boundary line of something. The key factors that decide my scopes are stated below -

1. Company overview and culture without violating the confidentiality
2. The training I have received
3. Growth of my technical experiences
4. My opinion on the internship

1.6 Limitations

There are also some limitations that are impactful on this report. In the following they are stated in brief:

1. Many parts of my internship experience are confidential by the global policies of Vivasoft Limited.

2. The report is based on my experience from 1 September 2021 to 30 March 2022. This is due to the demand from my university to submit before.

Internship is a source of experience for the future new comers to the industry. With the perfect utilization of it, the students can make a good impression about their starting careers. Also, it is a great scope for personal skill development.

Chapter 2

Company Profile

In this section, I am going to discuss about the environment and customs of **Vivasoft**.

2.1 Company overview

Vivasoft Limited is a Bangladeshi Software development company. Vivasoft is mainly a outsourcing company but now they are more focused to bring their own product on the market. Vivasoft built **PIHR**. PiHR is the best saas based HR and payroll management software in Bangladesh. Currently, It has 250+ clients. The services offering by vivasoft are:

- Software development
- ML/AI
- MVP As a Service
- Biz intelligence
- Mobile App Development

Its headquarter is located at Kamal Ataturk avenue, Banani, Dhaka. On September 2021 they reached their 5th year anniversary.

It has been growing so rapidly that it reaches 130 employee from 70 within a year

Figure 2.1: Employee growth

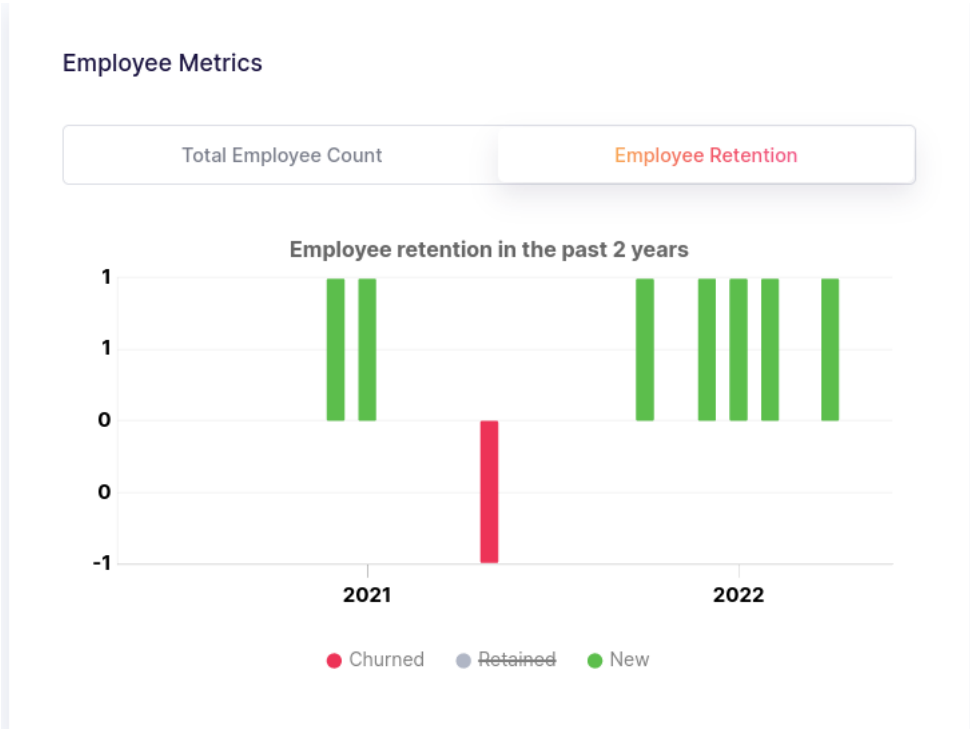


Figure 2.2: Employee Retention

2.2 Management

Vivasoft has two member in their managing board and they are also the founder of Vivasoft Limited. They jointly co-found the **Vivasoft limited**. Both of them are very friendly and easily accessible to their employees.

(a) Shafiul Hasan Tareq,
Managing Director Vivasoft Limited

(b) Shafqat Asif,
Co-founder Vivasoft Limited

Figure 2.3: Vivasoft Management

2.3 Location and Physical Layout

The headquarter of Vivasoft Limited is located on Ahmed Tower, Kemal Ataturk Ave, Dhaka 1213.

Figure 2.4: Office building

- Vivasoft office building is situated in an commercial area.
- The office consists of the 11th and 16th floor. Vivasoft is planning to expand its office area by including the 18th floor of the building.
- It has a indoor playground on 11th floor.
- It has a gymnasium zone with instruments where employees can exercise and keep their body fit.
- It has a nice dining room on the 11th floor of the building.
- And there is a big kitchen on 11th floor.

2.4 What Vivasoft Does

As I have mentioned earlier **Vivasoft** is a software development company. They help to bring the ideas of their clients to reality. They support the whole software development life cycle. They manage clients projects, consult, architect, design, develop and test. As

Vivasoft maintain continuous integration and continuous development/delivery after some sorts of development clients can view what they are developing and give their feedback. The services provided by vivasoft are:

- Software Development
- Software Quality Assurance
- ML/AI
- MVP As a Service
- Biz intelligence
- Mobile App Development

2.5 Vivasoft Products

- **PI-HR Solution**

  PI HR is the people management solution that helps in time of tracking, attendance, leave, salary management. PI-HR transform the way HR department works. Manage HR and payroll activities from a single software. The HR software is used by 250+ of companies across the country.

- **PI-TMS**

  Pi-TMS (Transport Management System) offers cost-effective web and mobile application based transport management software. Upgrade from paper bookkeeping and spreadsheet-based transport management to automated software in the cloud. Achieve preventive maintenance schedule automation, simpler fuel expense tracking, flexible reporting to improve your fleet management operations and decision making.

- **FLEET MANAGEMENT SOLUTION**

  Fleet (vehicle) management can include a range of functions, such as vehicle financing, vehicle maintenance, vehicle telematics (tracking and diagnostics), driver management, speed management, fuel management and health and safety management. We have integrated this solution in Gargaar Freighters Limited.

  Features:

  - Vehicle management.
  - Driver management.
  - Vendor Management.
  - Mobile app with GPS Tracking.
  - Invoice management.

- **MFS (Mobile Financing System)**

  Running in Somalia, MFS is a full fledged mobile wallet. Enabling the mass to do all sorts of financial operations possible using a platform like this, as a result, improving millions of lives.

  Features:

  - Cash In.
  - Cash Out.
  - Payment.
  - Send Money.
  - Interest.
  - Buy Airtime.

- **SELFCARE SOLUTIONS**

  Telesom is a private telecommunication company which was established in the year 2002 by local entrepreneurs in Hargeisa Somaliland. We have integrated the Selfcare product in Telesom company. We have integrated this solution in Telesom Selfcare.

  Features:

  - Mobile Apps
  - VAS Services
  - Cloud Service
  - Internet Packages Management
  - Vehicle Tracking System and many more

- **M-MARKET (Mobile Marketing)**

  M-market is a website where you can buy and sell almost everything. The best deals

are often done with people who live in your own city or on your own street, so on M-market it's easy to buy and sell locally. All you have to do is select your region.

Features:

- User can buy and sell almost everything.
- The best deals are often done with people who live in users own city or on users own street, so on Mmarket it's easy to buy and sell locally.
- It takes less than 2 minutes to post an ad on Mmarket.com.
- User can sign up for a free account and post ads easily every time.
- Mmarket.com does not specialize in any specific category.
- Vivasofft also carefully review all ads that are being published, to make sure the quality is up to their standards.

- **PI TICKETING**

  PI Ticketing is a online ticket platform. You can purchase your ticket through this platform. There are several types of ticketing systems are implemented in this platform. You can purchase your tickets through this platform.

  Features:

  - User can choose their origin, destination, journey dates and search for buses.
  - User can Select their desire trip and choose their seats.
  - Pay the tickets via Debit/Credit cards.
  - Safe and Secure online payments.
  - Cash on delivery.
  - Dependable customer service.

- **VIVA PENSION** Viva Pension is a product designed to cater the need of employers and employees in the national level. As its name suggests it is a product to handle the pension of a country.

  Features:

  - User Management
  - Employer

  - Notice
  - Member Management
  - Contribution
  - Investment
  - Expense
  - Profit Distribution
  - Chart of Account
  - Sales & Purchase
  - Withdrawal
  - Reports and many more

2.6 Tools and Frameworks

Over the years Vivasoft has worked with many tools and technology. Some of those are given below.

Languages

- C#
- Go
- Java
- Python
- Javascript
- Typescript
- Json
- YAML
- HTML
- CSS/SCSS/SASS

Frameworks

- ASP.NET Core
- ASP.NET MVC
- Angular
- Nextjs
- Nestjs
- Echo
- Spring boot
- Fast API

Databases

- MongoDB
- Mysql
- Postgres
- MSSQL Server

Servers

- Nginx

Cloud

- Amazon Web Service
- Microsoft Azure
- Digital ocean

DevOps Tools

- Docker
- Kubernetes
- Docker Swarm
- Helm
- Jenkins
- Terraform
- Tsung

2.7 Domestic Environment

The most interesting thing about Vivasoft is its work environment. Vivasoft maintains a friendly environment and almost all the employees are happy with such a friendly environment.

2.8 Lunch and snacks

The employees are provided with lunch and snacks from the office. Also, there is tea, coffee, biskuits available all the time.

2.9 Recreation

Vivasoft has different ways of recreation of employees. Arranging indoor games tournament, participating Inter Software Sports events etc. Employees are taken out of the country for the international tour every year.

Figure 2.5: Badminton tournament

Figure 2.6: India tour

Figure 2.7: Cricket Practice at Mirpur DOHS Central field

Figure 2.8: Vivasoft Cricket team

Figure 2.9: Iftar mahfil

Figure 2.10: Iftar mahfil

Chapter 3

Internship Activites

The course of journey of an intern is well structured at Vivasoft Limited. Vivasoft Limited believes in learning by doing. Vivasoft has their own training plan to train interns. Course plans are different for different team. As I was on **DevOps** team my intern activities were different from others.

I can discuss my internship period in two parts:

1. Things I learned at Vivasoft.
2. Things I did at Vivasoft.

3.1 Things I learned at Vivasoft

3.1.1 Golang

As I was on **DevOps** team I didn't have to write code. But if I have the knowledge about the stacks I am going to deploy it will help me to better understand the project . The first project I worked at Vivasoft was written in **Golang**. That's why I had to learn basic syntex of **Glang**. The topics I have gone through at Vivasoft are:

- Packages, Variables, Functions
- Flow control statement
- structs, slices, maps

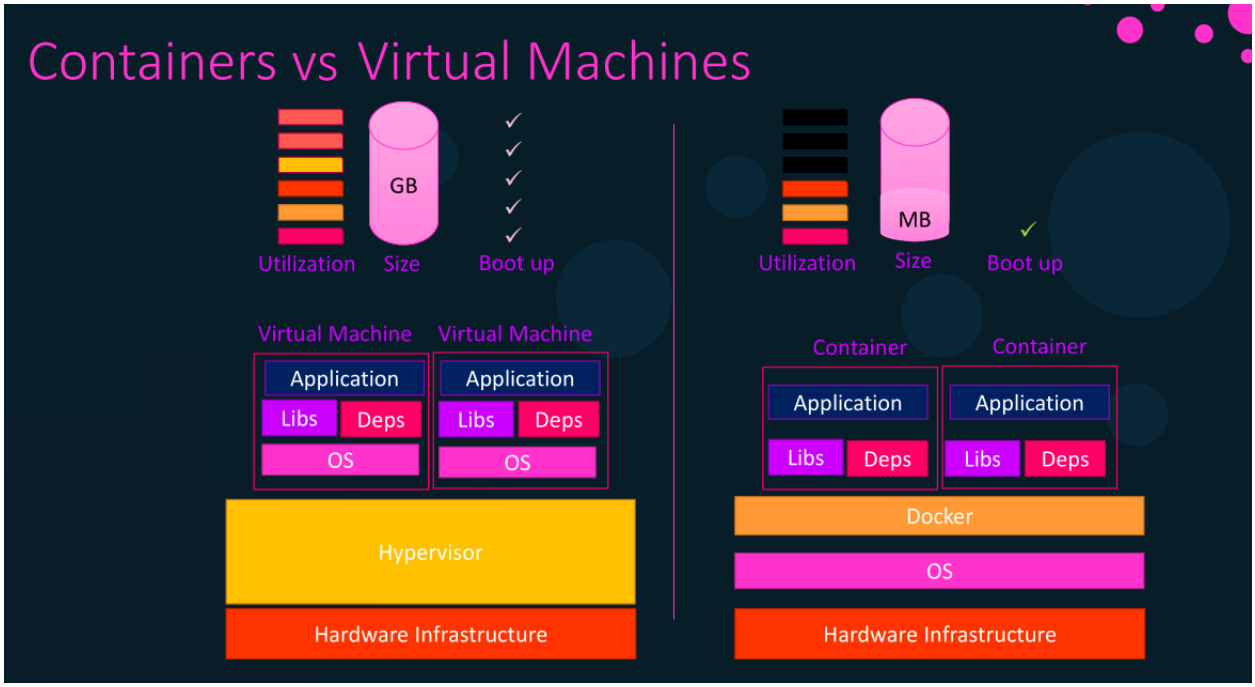
- Methods, Interfaces, Pointer
- Concurrency, go routine etc.

3.1.2 Containerization

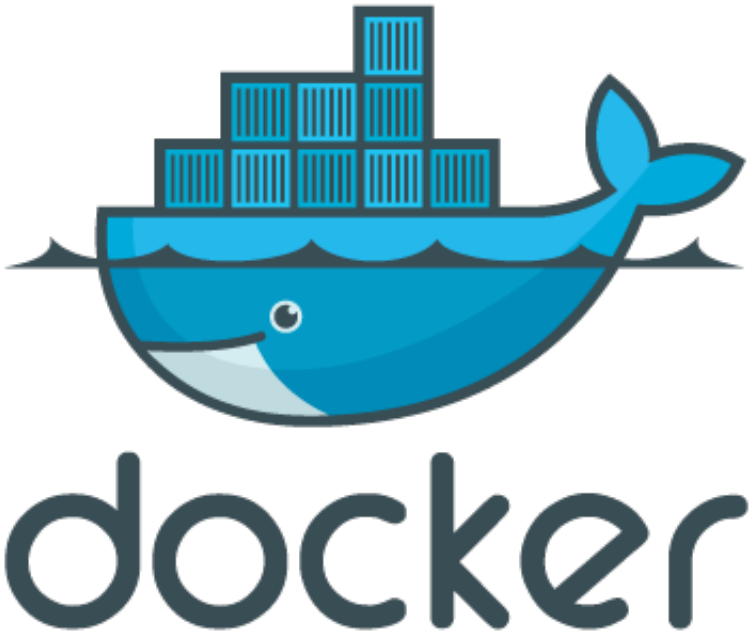
Containerization is a form of virtualization where applications run in isolated user spaces, called containers, while using the same shared operating system (OS). One of the benefits of containerization is that a container is essentially a fully packaged and portable computing environment.

Everything an application needs to run—its binaries, libraries, configuration files, and dependencies—is encapsulated and isolated in its container. The container itself is abstracted away from the host OS, with only limited access to underlying resources—much like a lightweight virtual machine (VM). As a result, the containerized application can be run on various types of infrastructure—on bare metal, within VMs, and in the cloud—without needing to refactor it for each environment.

With containerization technology, there's less overhead during startup and no need to set up a separate guest OS for each application since they all share the same OS kernel. Because of this high efficiency, containerization is commonly used for packaging up the many individual micros-ervices that make up modern apps.



3.1.3 Docker



As a part of learning container run time engine I have learned **Docker**. **Docker** is the most popular container run time engine at this moment.

Docker is an open source project for building, shipping, and running programs. It is a command-line program, a background process, and a set of remote services that take a logistical approach to solving common software problems and simplifying our experience installing, running, publishing, and removing software. It accomplishes this by using an operating system technology called **containers**.

Docker is so popular because it enhanced the native Linux containerization capabilities with technologies that enable:
Improved—and seamless—portability: While LXC containers often reference machine-specific configurations, Docker containers run without modification across any desktop, data center and cloud environment.

Even lighter weight and more granular updates: With LXC, multiple processes can be combined within a single container. With Docker containers, only one process can

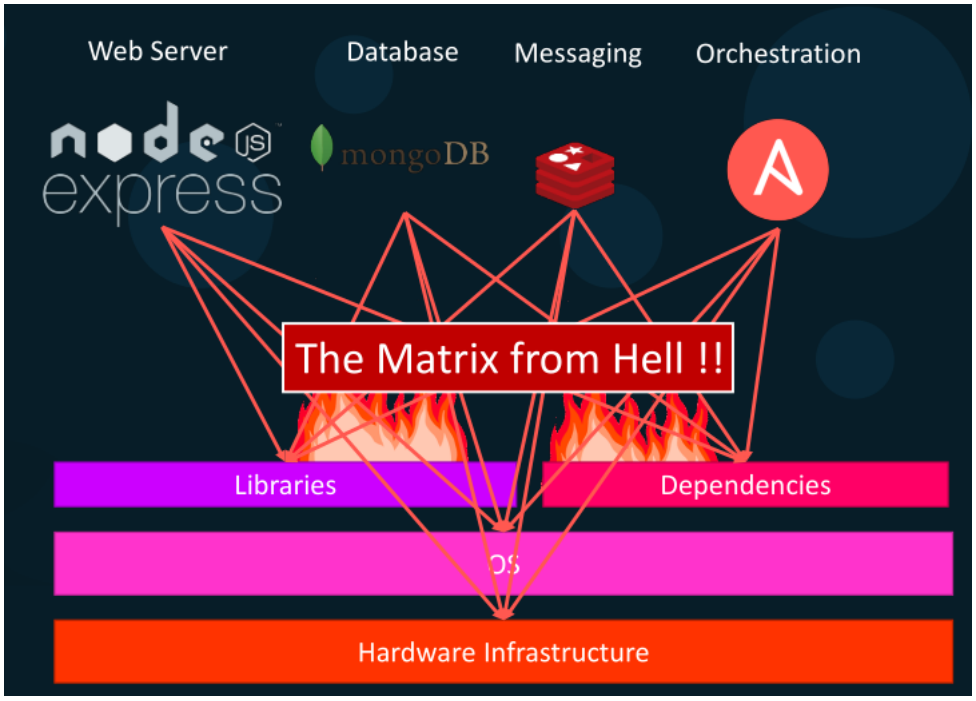
run in each container. This makes it possible to build an application that can continue running while one of its parts is taken down for an update or repair.

Automated container creation: Docker can automatically build a container based on application source code.

Container versioning: Docker can track versions of a container image, roll back to previous versions, and trace who built a version and how. It can even upload only the deltas between an existing version and a new one.

Container reuse: Existing containers can be used as base images—essentially like templates for building new containers.

Shared container libraries: Developers can access an open-source registry containing thousands of user-contributed containers



Docker solved **The Matrix from Hell** and some other problems like:

- Compatibility/Dependency
- Long setup time

- Different Dev/Test/Prod environments
- Containerize Applications
- Run each service with its own dependencies in separate containers

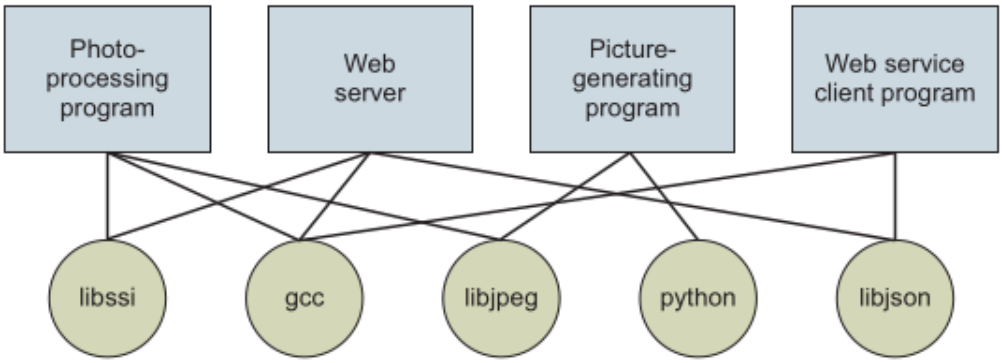


Figure 3.1: Dependency relationships without docker

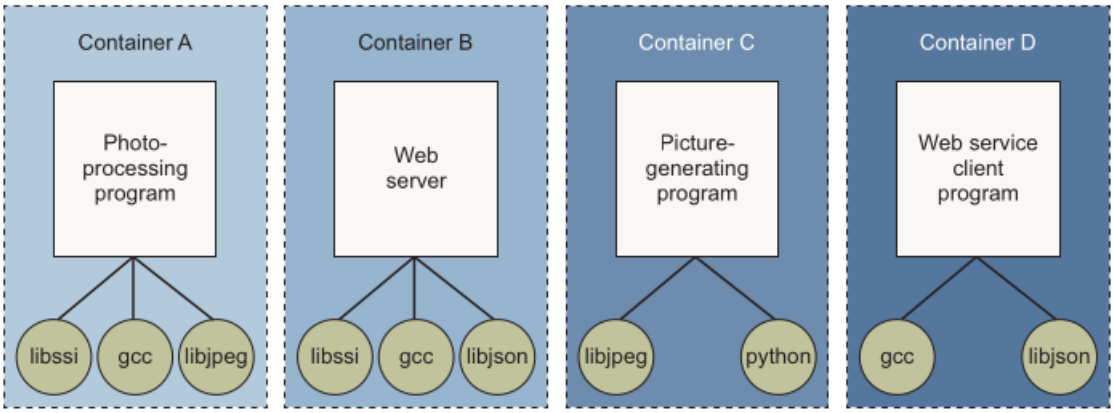


Figure 3.2: Dependency relationships without docker

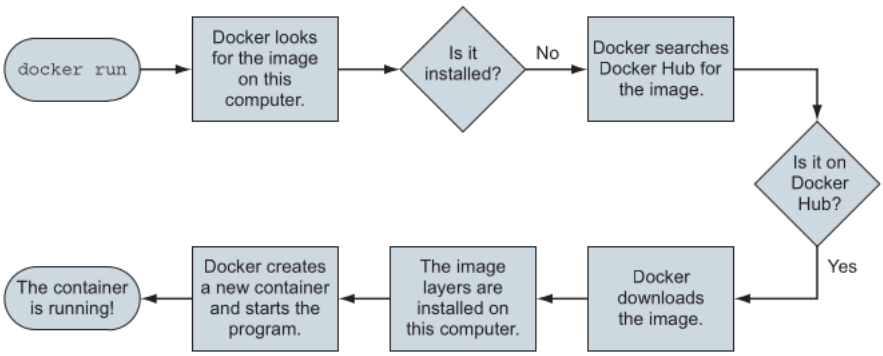


Figure 3.3: What happens after running docker run

3.2 Kubernetes

Kubernetes is a container management tool or container orchestration engine. It automatically manages, scales, and maintains multi-container workloads in desired states.

Modern software is increasingly run as fleets of containers, sometimes called microservices. A complete application may comprise many containers, all needing to work together in specific ways. Kubernetes is software that turns a collection of physical or virtual hosts (servers) into a platform that:

- Hosts containerized workloads, providing them with compute, storage, and network resources, and
- Automatically manages large numbers of containerized applications — keeping them healthy and available by adapting to changes and challenges.

The tasks done by kubernetes are:

- Automatic bin packing

- Service discovery
- Storage orchestration
- Self healing
- Deploying
- Scheduling
- Scaling
- Load balancing
- Batch execution
- Rollbacks
- Monitoring

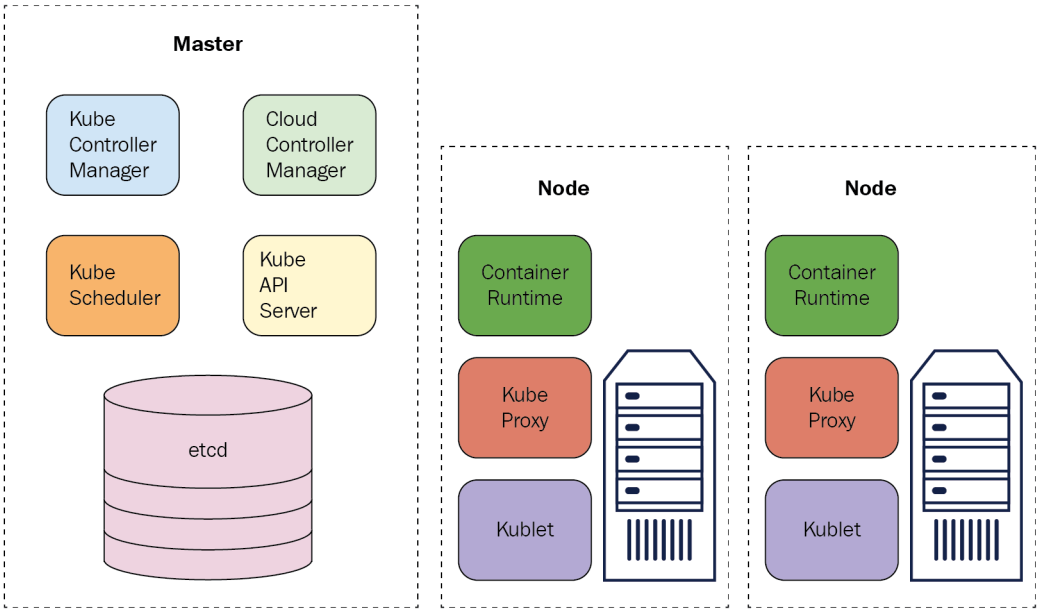
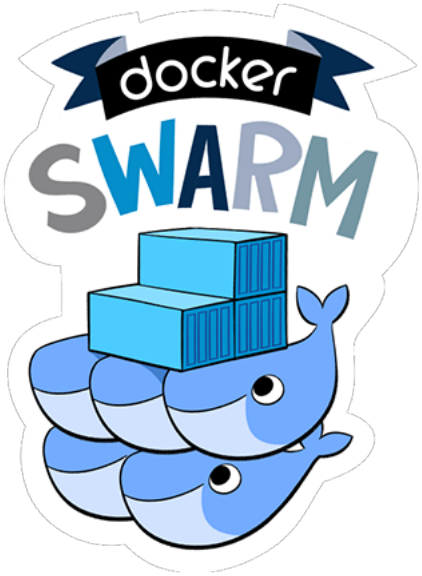


Figure 3.4: Kubernetes cluster

3.3 Docker Swarm

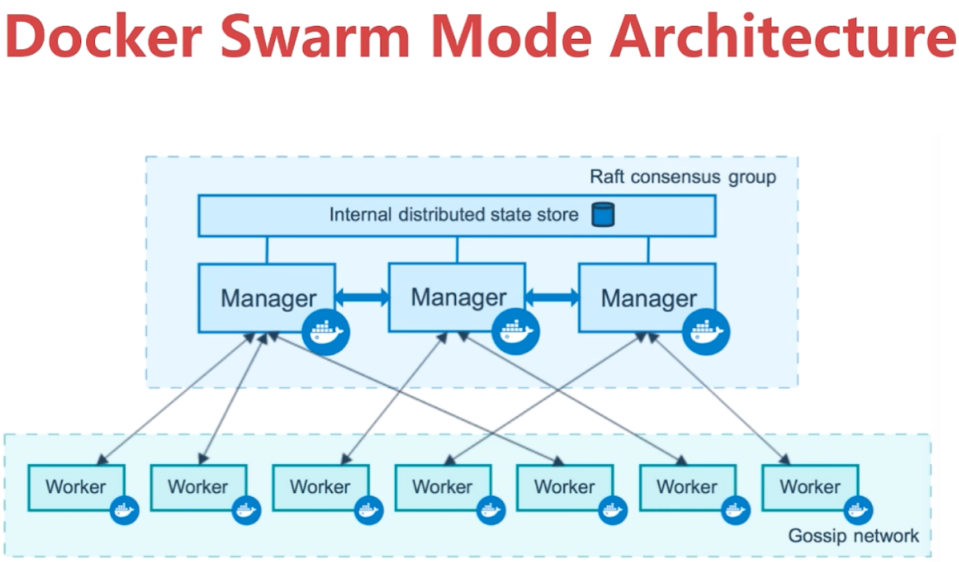


A **Docker Swarm** is a group of either physical or virtual machines that are running the Docker application and that have been configured to join together in a cluster. Once a group of machines have been clustered together, you can still run the Docker commands that you're used to, but they will now be carried out by the machines in your cluster. The activities of the cluster are controlled by a **swarm manager**, and machines that have joined the cluster are referred to as **nodes**.

Docker swarm is a container **orchestration** tool, meaning that it allows the user to manage multiple containers deployed across multiple host machines.

One of the key benefits associated with the operation of a docker swarm is **the high level of availability** offered for applications. In a docker swarm, there are typically several worker nodes and at least one manager node that is responsible for handling the worker nodes' resources efficiently and ensuring that the cluster operates efficiently.

Docker Swarm does **load balancing**. Docker Swarm's load balancer runs on every node and is capable of balancing load requests across multiple containers and hosts.



3.4 Helm

Helm is a tool for managing Charts. Charts are packages of pre-configured Kubernetes resources.

Helm is a tool that streamlines installing and managing Kubernetes applications. We can think of it like apt/yum/homebrew for Kubernetes.

- Helm renders users templates and communicates with the Kubernetes API
- Helm runs on users laptop, CI/CD, or wherever we want it to run.
- Charts are Helm packages that contain at least two things:
  - A description of the package (Chart.yaml)
  - One or more templates, which contain Kubernetes manifest files
- Charts can be stored on disk, or fetched from remote chart repositories (like Debian or RedHat packages)

3.5 Terraform

Terraform is an infrastructure as code (IaC) tool that allows users to build, change, and version infrastructure safely and efficiently. This includes both low-level components like compute instances, storage, and networking, as well as high-level components like DNS entries and SaaS features.

Example code to provision a **EC2** machine from **AWS**

```
resource "aws_instance" "gps_consul" {
  ami           = data.aws_ami.amazon-linux-2.id
  instance_type = var.instance_type
  vpc_security_group_ids = [
    aws_security_group.consul_sg.id
  ]
  key_name  = var.gps_service_access_key
  user_data = file("./scripts/post-consul-operations.sh")
  tags = {
    Name = "consul"
  }
}
```

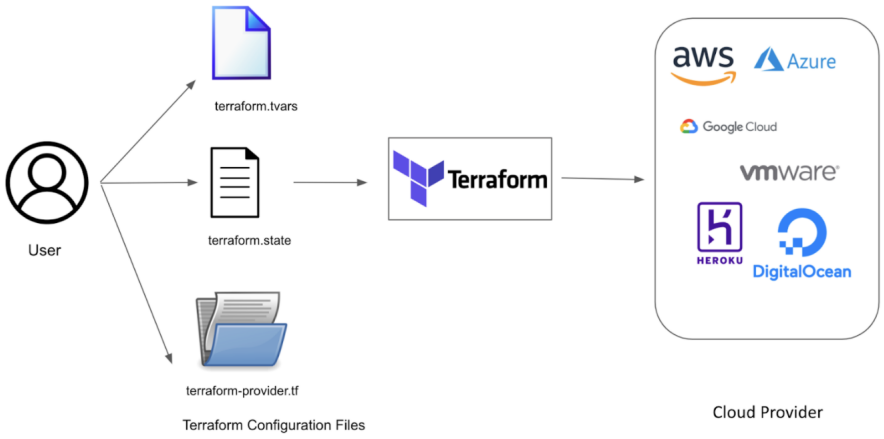


Figure 3.5: Terraform work flow

3.6 Tsung

Tsung (formerly IDX-Tsunami) is a distributed **load testing** tool. It is protocol-independent and can currently be used to stress HTTP, WebDAV, SOAP, PostgreSQL, MySQL, AMQP, MQTT, LDAP and Jabber/XMPP servers.

Tsung's main strength is its ability to simulate a huge number of simultaneous user from a single machine; moreover, we can distribute the users on cluster for machines. When used on cluster, we can generate a really impressive load on a server with a modest cluster, easy to set-up and to maintain. It is possile to use Tsung on a cloud like EC2.

Tsung is developed in **Erlang** and this is where the power of Tsung resides.

Erlang is a **concurrency-oriented** programming language. Tsung is based on the Erlang OTP (Open Telecom Platform) and inherits several characteristics from Erlang:

Performance

Erlang has been made to support hundred thousands of lightweight processes in a single virtual machine.

Scalability

Erlang runtime environment is naturally distributed, promoting the idea of process's location transparency.

Fault-tolerance

Erlang has been built to develop robust, fault-tolerant systems. As such, wrong answer sent from the server to Tsung does not make the whole running benchmark crash.

3.7 Jenkins

Jenkins is a self-contained, open source automation server which can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software.

Jenkins offers a simple way to set up a **continuous integration or continuous delivery (CI/CD)** environment for almost any combination of languages and source code repositories using pipelines, as well as automating other routine development tasks. While Jenkins doesn't eliminate the need to create scripts for individual steps, it does give us a faster and more robust way to integrate our entire chain of build, test, and deployment tools than we can easily build ourselves.

A sample pipeline script:

```
pipeline {
    agent any
    stages {
        stage("Build") {
            steps {
                echo "Building.."
            }
        }
        stage("Test") {
            steps {
                echo "Testing.."
            }
        }
        stage("Deploy") {
            steps {
                echo "Deploying...."
            }
        }
    }
}
```

pipeline is the mandatory outer block to invoke the Jenkins pipeline plugin. agent defines where we want to run the pipeline. any says to use any available agent to run the pipeline or stage. A more specific agent might declare a **container** to use, for example:

```
agent {
    docker {
        image "maven:3-alpine"
        label "my-defined-label"
```

```
        args  "-v /tmp:/tmp"
    }
}
```

stages contain a sequence of one or more stage directives. In the example above, the three stages are Build, Test, and Deploy.

steps do the actual work. In the example above the steps just printed messages. A more useful build step might look like the following:

```
pipeline {
    agent any

    stages {
        stage("Build") {
            steps {
                docker build  -t my-app .
            }
        }
    }
}
```

3.8 Amazon Web Service

AWS (Amazon Web Services) is a comprehensive, evolving cloud computing platform provided by Amazon that includes a mixture of **infrastructure as a service (IaaS)**,

platform as a service (PaaS) and **packaged software as a service (SaaS)** offerings. **AWS** services can offer an organization tools such as compute power, database storage and content delivery services.
AWS was one of the first companies to introduce a **pay-as-you-go** cloud computing model that scales to provide users with compute, storage or throughput as needed.

As of 2021, AWS comprises over 200 products and services including computing, storage, networking, database, analytics, application services, deployment, management, machine learning, mobile, developer tools, RobOps and tools for the Internet of Things.

The services I have used during my internship period are:

3.8.1 Route53

Amazon Route 53 is a highly available and scalable **Domain Name System (DNS)** web service. It is designed for developers and corporates to route the end users to Internet applications by translating human readable names like www.mydomain.com, into the numeric IP addresses like 192.0.2.1 that computers use to connect to each other.

3.8.2 EC2

Amazon EC2 (Elastic Compute Cloud) is a web service interface that provides resizable compute capacity in the AWS cloud. It is designed for developers to have complete control over web-scaling and computing resources.

EC2 instances can be resized and the number of instances scaled up or down as per our requirement. These instances can be launched in one or more geographical locations or regions, and **Availability Zones (AZs)**. Each region comprises of several AZs at distinct locations, connected by low latency networks in the same region.
Features of EC2
Here is a list of some of the prominent features of EC2 -

- **Reliable** - Amazon EC2 offers a highly reliable environment where replacement of instances is rapidly possible. Service Level Agreement commitment is 99.9

- **Designed for Amazon Web Services** - Amazon EC2 works fine with Amazon services like Amazon S3, Amazon RDS, Amazon DynamoDB, and Amazon SQS. It provides a complete solution for computing, query processing, and storage across a wide range of applications.

- **Secure** - Amazon EC2 works in Amazon Virtual Private Cloud to provide a secure and robust network to resources.

- **Flexible Tools** - Amazon EC2 provides the tools for developers and system administrators to build failure applications and isolate themselves from common failure situations.

- **Inexpensive** - Amazon EC2 wants us to pay only for the resources that we use. It includes multiple purchase plans such as On-Demand Instances, Reserved Instances, Spot Instances, etc. which we can choose as per our requirement.

3.8.3 VPC

Amazon Virtual Private Cloud (VPC) allows the users to use AWS resources in a virtual network. The users can customize their virtual networking environment as they like, such as selecting own IP address range, creating subnets, and configuring route tables and network gateways.

3.8.4 RDS

Amazon RDS (Relational Database Service) is a fully-managed SQL database cloud service that allows to create and operate relational databases. Using RDS you can access your files and database anywhere in a cost-effective and highly scalable way.

Features of Amazon RDS

Amazon RDS has the following features -

- **Scalable** - Amazon RDS allows to scale the relational database by using AWS Management Console or RDS-specific API. We can increase or decrease your RDS requirements within minutes.

- **Host replacement** - Sometimes these situations occur when the hardware of Amazon RDS fails. There is no need to worry, it will be automatically replaced by Amazon.
- **Inexpensive** - Using Amazon RDS, we pay only for the resources we consume. There is no up-front and long-term commitment.
- **Secure** - Amazon RDS provides complete control over the network to access their database and their associated services.
- **Automatic backups** - Amazon RDS backs up everything in the database including transaction logs up to last five minutes and also manages automatic backup timings.
- **Software patching** - Automatically gets all the latest patches for the database software. We can also specify when the software should be patched using DB Engine Version Management.

3.8.5 DynamoDB

Amazon DynamoDB is a fully managed NoSQL database service that allows to create database tables that can store and retrieve any amount of data. It automatically manages the data traffic of tables over multiple servers and maintains performance. It also relieves the customers from the burden of operating and scaling a distributed database. Hence, hardware provisioning, setup, configuration, replication, software patching, cluster scaling, etc. is managed by Amazon.

Benefits of Amazon DynamoDB

Managed service - Amazon DynamoDB is a managed service. There is no need to hire experts to manage NoSQL installation. Developers need not worry about setting up, configuring a distributed database cluster, managing ongoing cluster operations, etc. It handles all the complexities of scaling, partitions and re-partitions data over more machine resources to meet I/O performance requirements.

Scalable - Amazon DynamoDB is designed to scale. There is no need to worry about

predefined limits to the amount of data each table can store. Any amount of data can be stored and retrieved. DynamoDB will spread automatically with the amount of data stored as the table grows.

Fast - Amazon DynamoDB provides high throughput at very low latency. As datasets grow, latencies remain stable due to the distributed nature of DynamoDB's data placement and request routing algorithms.

Durable and highly available - Amazon DynamoDB replicates data over at least 3 different data centers' results. The system operates and serves data even under various failure conditions.

Flexible - Amazon DynamoDB allows creation of dynamic tables, i.e. the table can have any number of attributes, including multi-valued attributes.

Cost-effective - Payment is for what we use without any minimum charges. Its pricing structure is simple and easy to calculate.

3.8.6 S3

Amazon S3 (Simple Storage Service) is a scalable, high-speed, low-cost web-based service designed for online backup and archiving of data and application programs. It allows to upload, store, and download any type of files up to 5 TB in size. This service allows the subscribers to access the same systems that Amazon uses to run its own web sites. The subscriber has control over the accessibility of data, i.e. privately/publicly accessible.

3.8.7 CloudFront

CloudFront is a **CDN (Content Delivery Network)**. It retrieves data from Amazon S3 bucket and distributes it to multiple datacenter locations. It delivers the data through a network of data centers called edge locations. The nearest edge location is routed when the user requests for data, resulting in lowest latency, low network traffic, fast access to data, etc.

Features of CloudFront

Fast: The broad network of edge locations and CloudFront caches copies of content close to the end users that results in lowering latency, high data transfer rates and low network traffic. All these make CloudFront fast.

Simple: It is easy to use.

Can be used with other AWS Services: Amazon CloudFront is designed in such a way that it can be easily integrated with other AWS services, like Amazon S3, Amazon EC2.

Cost-effective: Using Amazon CloudFront, we pay only for the content that you deliver through the network, without any hidden charges and no up-front fees.

Elastic: Using Amazon CloudFront, we need not worry about maintenance. The service automatically responds if any action is needed, in case the demand increases or decreases.

Reliable: Amazon CloudFront is built on Amazon's highly reliable infrastructure, i.e. its edge locations will automatically re-route the end users to the next nearest location, if required in some situations.

Global: Amazon CloudFront uses a global network of edge locations located in most of the regions.

3.8.8 IAM

AWS Identity and Access Management (IAM) provides fine-grained access control across all of AWS. With IAM, user can specify who can access which services and resources, and under which conditions. With IAM policies, user manage permissions to your workforce and systems to ensure least-privilege permissions.

IAM is an AWS service that is offered at no additional charge

3.8.9 ACM

AWS Certificate Manager is a service that lets us easily provision, manage, and deploy public and private Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for use with AWS services and our internal connected resources. SSL/TLS certificates are used to secure network communications and establish the identity of websites over the Internet as well as resources on private networks. AWS Certificate Manager removes the time-consuming manual process of purchasing, uploading, and renewing SSL/TLS certificates.

3.8.10 Codepipeline

AWS CodePipeline is a fully managed continuous delivery service that helps users automate their release pipelines for fast and reliable application and infrastructure updates. CodePipeline automates the build, test, and deploy phases of users release process every time there is a code change, based on the release model user define. This enables user to rapidly and reliably deliver features and updates. User can easily integrate AWS CodePipeline with third-party services such as GitHub or with users own custom plugin. With AWS CodePipeline, user only pay for what they use. There are no upfront fees or long-term commitments.

3.8.11 ECS

Amazon Elastic Container Service (Amazon ECS) is a highly scalable, fast, container management service that makes it easy to run, stop, and manage Docker containers on a cluster of Amazon EC2 instances.

3.8.12 EKS

Amazon Elastic Kubernetes Service (Amazon EKS) is a managed container service to run and scale Kubernetes applications in the cloud or on-premises.

3.8.13 ECR

Amazon ECR is a fully managed container registry offering high-performance hosting, so user can reliably deploy application images and artifacts anywhere.

3.9 Non Technical Aspects

Professionalism

Professionalism is the conduct, behavior and attitude of someone in a work or business environment. The environment in office and in university is quite different. I have learned to take ownership of what I am doing. Software development is always is teamwork. So the whole team is responsible for any achievements and failure. There is no scope to blame individual.

Knowledge Sharing

Knowledge sharing is the exchange of information or understanding between people, teams, communities, or organizations. In Vivasoft, I could ask question to my mentor any time, everyone is so friendly that I was able to ask question everyone and they are very helpful. My mentor always encourage me to discussion. Though I was an intern he told me both of us can learn from each other.

Communication

Communication skill is very important for everyone. Every software professional should be good at it because we always work in a team. In Vivasoft, I learned how to approach to people by following my senior, how politely they approach.

Planning

Planning is the most important thing to do a job efficiently. In Vivasoft, my senior always encourage me to plan before starting execution.

3.10 Things I did at Vivasoft

3.10.1 GPS Server Deployment

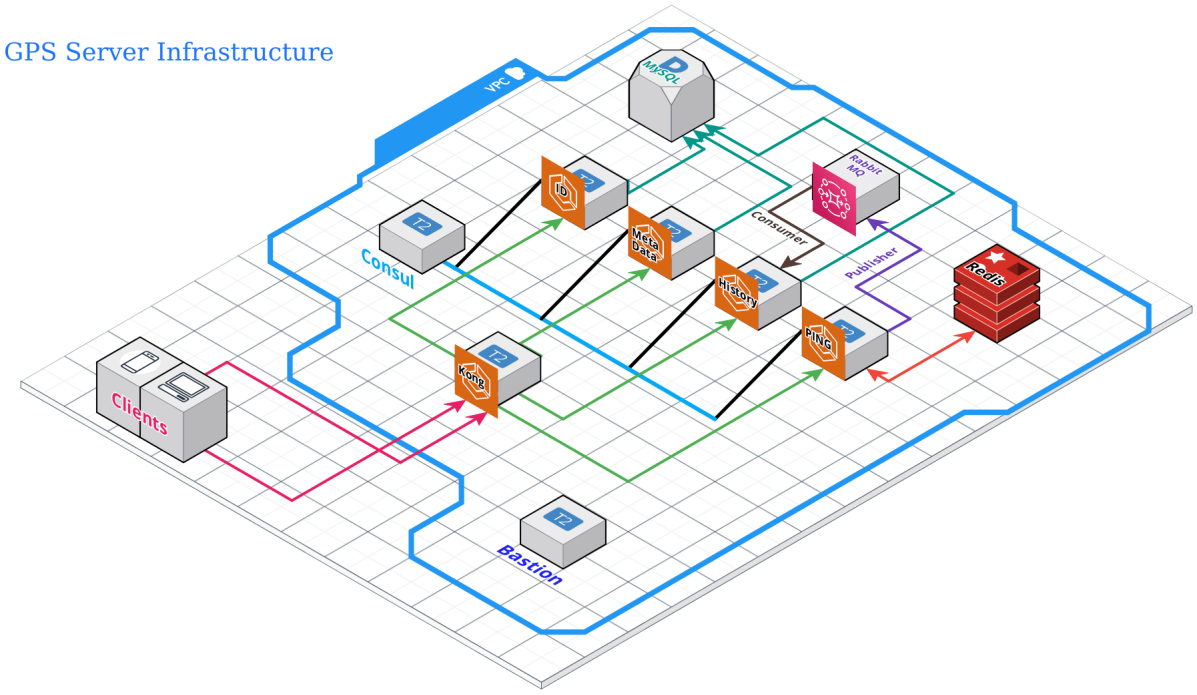


Figure 3.6: GPS Server Infrastructure

The figure is a bird's eye view of the **GPS server infrastructure**. It is developed using **mirco-service** architecture. There are **four services** in the GPS server and they are **Ping**, **History**, **Id**, and **Metadata**. Clients or users can only access the services through the **Kong Gateway**. There is an instance for configuration management named **Consul**. All the service has access to it. Before starting any service they get their configuration from the Consul instance. There is also a **Rabbitmq** for implementing message queue, **Redis for cache/in-memory database**, and a **Mysql** database to store data. Ping service is the publisher for Rabbitmq. All the user location data first goes to Ping service and Ping service pushes them into Rabbitmq and only the last location of each user is cached on Redis. History Service is the consumer of Rabbitmq. It consumes data from the message queue and stores them in the Mysql database. Id service is used for authen-

tication of the user.

Services

GPS Server is written in **Golang**. Simple description of the services are given below

Ping

The service saves all the locations of a user. At first, it stores the location data on memory and after a certain time, it pushes them to the Rabbitmq as a bulk. For every user, it stores the last location of users to the Redis cache. There are four endpoints in this service. They are

1. **GET /api/v1/ping**: This endpoint is for checking the health of the service.
2. **POST /api/v1/locations**: This endpoint saves all the user data.
3. **GET /api/v1/locations/users**: This endpoint is for retrieving all users within a specific radius.
4. **GET /api/v1/locations**: Get the last location of specific users.

History

The service provides all the location histories of a user. It consumes users' location data from rabbitmq and pushes them into the database. This service is the only consumer of the rabbitmq. There are two endpoints in this service. They are:

1. **GET /api/v1/ping**: This endpoint is for checking the health of the service.
2. **GET /api/v1/history**: The endpoint will provide all the location histories of a specific user.

Id

The service manages authentication and authorization. The service has five endpoints and they are:

1. **GET /api/v1/ping**: This endpoint is for checking the health of the service.
2. **GET /api/v1/internal/authenticate**: This endpoint is only for Gateway service's internal use.
3. **POST /api/v1/company/signup**: Endpoint to register a company. This is a restricted endpoint. This will not be provided to any person outside of the Realtime Location Service. All requests must have a sret header.
4. **POST /api/v1/user/signup**: Endpoint to register users for the companies.
5. **GET /api/v1/user/resolve**: Endpoint to retrieve all users for a company/AppKey.

Metadata

Metadatas are dynamic variable related to an entity. We assumed the entity is a user. The endpoints served by the service are :

1. **GET /api/v1/ping**: The endpoint is to check the health of the service.
2. **POST /api/v1/users/meta**: Create/Replace User.
3. **PATCH /api/v1/users/userId/meta**: Merge the given user metadata with existing value.
4. **GET /api/v1/users**: Returns all the users page by page.
5. **POST /api/v1/users/meta/search**: Gets users satisfying the query. There are three supported types of queries available. **MATCH**, **EXCEPT**, and **RANGE**.

Gateway

The service is the Gateway of all services. All public endpoints have to access through the gateway. Restricted endpoints can not be accessed through the gateway, they can be accessed another way

Cloud Solution

The whole infrastructure is deployed on AWS. All the services along with Gateway is deployed on docker container using **ECS**.

IAC

IAC stands for Infrastructure as code. **Terraform** is used here as an IAC tool. The whole infrastructure is written as terraform configuration for provisioning AWS resources. When we have to deal with a large infrastructure it is very handy to use an IAC tool. Among lots of IAC tools, we choose to use **terraform** because it supports all the Cloud Providers and as we just need an IAC tool for provisioning resources **terraform** is best fitted to our requirement.

Request Handle

Main Statistics

Name	highest 10sec mean	lowest 10sec mean	Highest Rate	Mean Rate	Mean	Count
connect	8.29 msec	6.58 msec	22.7 / sec	19.28 / sec	7.51 msec	11870
page	68.33 msec	4.72 msec	2170.3 / sec	1933.40 / sec	25.72 msec	1187000
request	68.33 msec	4.72 msec	2170.3 / sec	1933.40 / sec	25.72 msec	1187000
session	7.26 sec	0.50 sec	25.3 / sec	19.37 / sec	2.61 sec	11870

Transactions Statistics

Name	highest 10sec mean	lowest 10sec mean	Highest Rate	Mean Rate	Mean	Count
tr_post_request	68.69 msec	4.97 msec	2170.2 / sec	1933.40 / sec	25.99 msec	1187000

Network Throughput

Name	Highest Rate	Total
size_rcv	9.57 Mbits/sec	653.97 MB
size_sent	7.18 Mbits/sec	491.29 MB

Counters Statistics

Name	Highest Rate	Mean Rate	Total number

Name	Max
connected	144
finish_users_count	11870
users	166
users_count	11870

Server monitoring

Name	highest 10sec mean	lowest 10sec mean
cpu:os_mon@tsung_controller@tsung-master-0.tsung-master.load-test.svc.cluster.local	27.44 %	16.85 %
freemem:os_mon@tsung_controller@tsung-master-0.tsung-master.load-test.svc.cluster.local	6790.69 MB	6765.04 MB
load:os_mon@tsung_controller@tsung-master-0.tsung-master.load-test.svc.cluster.local	2.25	0.29

HTTP return code

Code	Highest Rate	Mean Rate	Total number
200	2170.3 / sec	1933.40 / sec	1187000

Figure 3.7: Load test report

Currently, It can serve 1500 request per second.

NOTE:

As Vivasoft open the GPS server for all that's why I have written details about it

3.10.2 Wellteam Deployment

Wellteam is a project to encourage people maintaining healthy life style. It arranges some challenge for its user. **Wellteam** backend is developed using micro-service architecture and **Golang** and front-end is developed using **Nextjs**. There are eight services in wellteam back-end they are

1. Auth service
2. Blog service
3. Core service
4. Email service
5. Survey service
6. Subscription service
7. File-uploader service
8. And Gateway service

All the service endpoints are private except **Gateway** that means end-user can access them through the **Gateway** service. **Gateway** service is behind a loadbalancer so user can access the **Gateway** service through the loadbalancer.

Cloud Solution

Wellteam is deployed on AWS(Amazon Web Service). Resources provisioned from AWS are:

- 3 EC2 instance
- 1 Application loadbalancer
- 1 S3 bucket
- SSL certificates from AWS ACM
- Hosted zone on Route53

Mysql database deployed on a dedicated EC2 instance as a docker container.

Container Management Tool

All the services along with front-end is deployed on **Docker Swarm**. There is only master node in the cluster so all the services are deployed on master node.

CI/CD tool

Jenkins is deployed on a dedicated EC2 instance as a continuous integration and continuous deployment/delivery tool.

S	W	Name ↓	Last Success	Last Failure	Last Duration	Fav
		blog	7 days 22 hr #8	15 days #7	2 min 19 sec	
		Clean Images	N/A	N/A	N/A	
		core	7 days 21 hr #37	7 days 21 hr #36	2 min 45 sec	
		email	1 mo 2 days #8	15 days #9	2 min 29 sec	
		file_uploader	1 mo 2 days #6	15 days #7	2 min 51 sec	
		gateway	1 mo 13 days #1	15 days #2	3 min 51 sec	
		subscription	7 days 21 hr #10	15 days #7	2 min 17 sec	
		survey	7 days 22 hr #10	15 days #9	2 min 20 sec	
		sync-consul-dev	14 days #6	N/A	8.1 sec	
		sync-consul-prod	14 days #9	15 days #2	3.2 sec	
		user	7 days 21 hr #14	7 days 21 hr #13	2 min 14 sec	
		wellteam-frontend	14 days #2	N/A	28 sec	

Figure 3.8: Jenkins pipelines

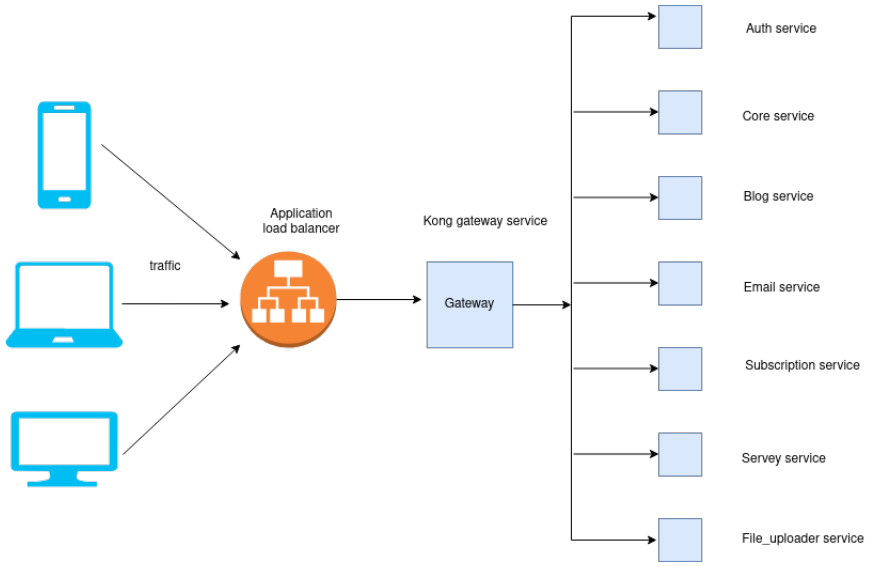
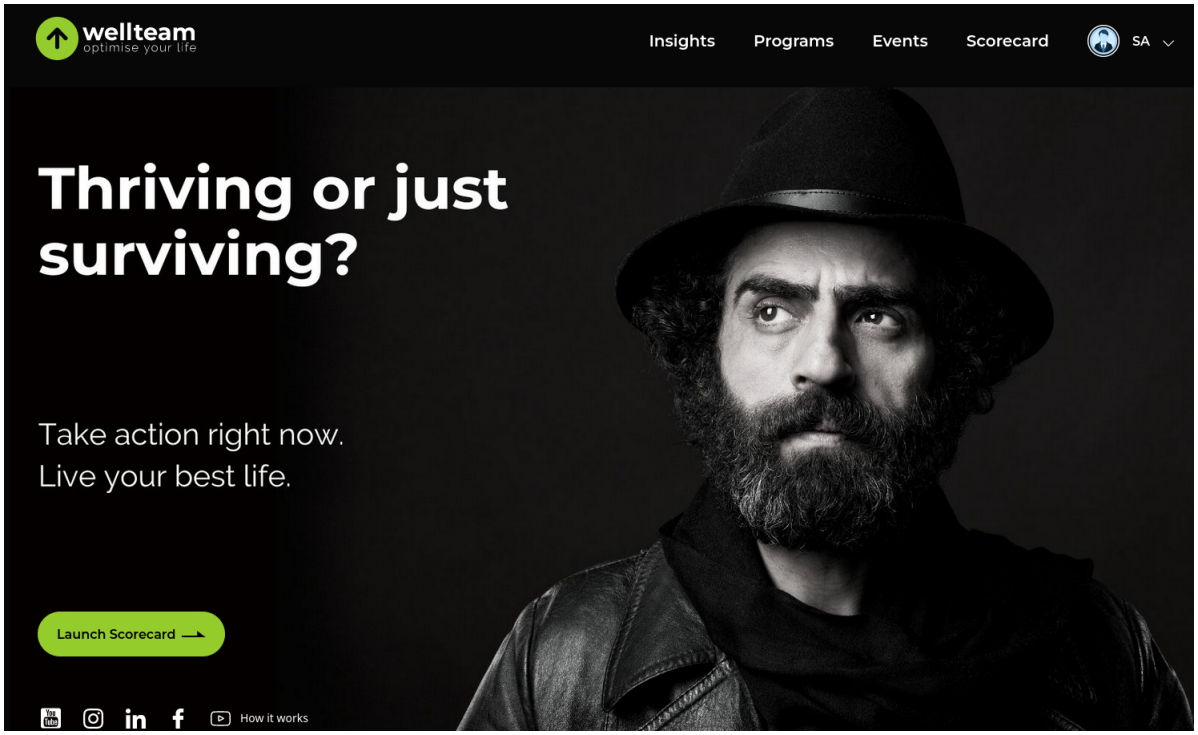
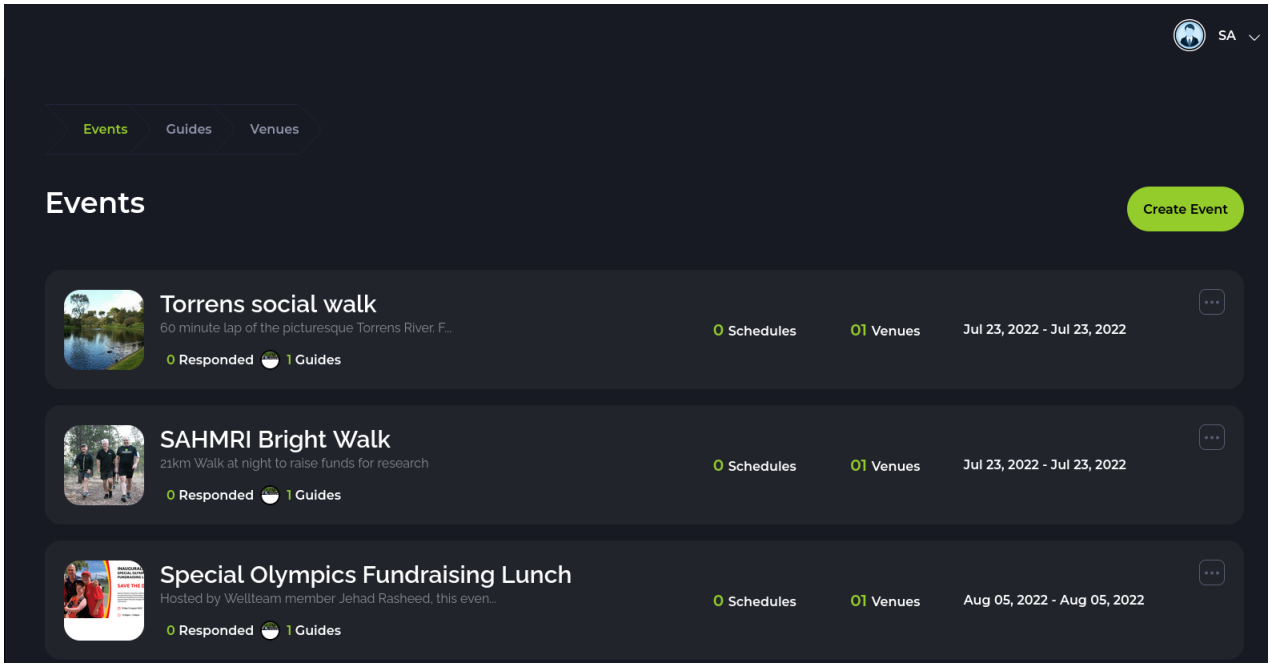
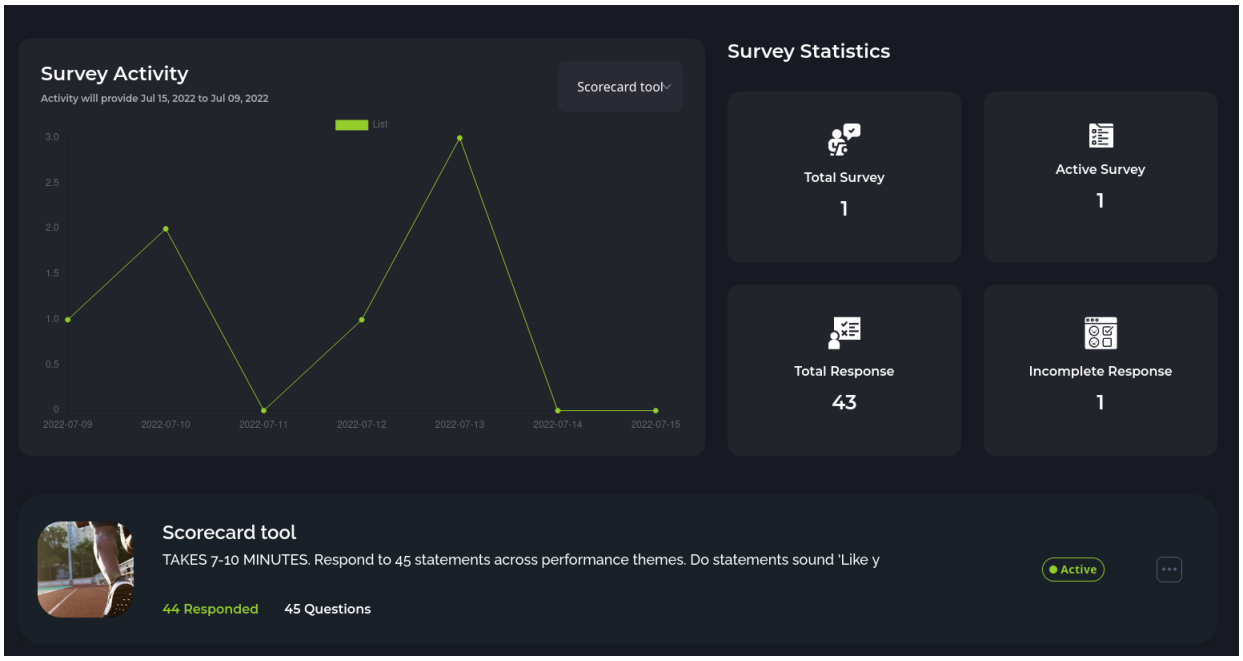


Figure 3.9: wellteam traffic flow

3.10.3 Some screen shots




SA
Events
Guides
Venues
Events
Create Event
Torrens social walk
60 minute lap of the picturesque Torrens River. F...
0 Responded
1 Guides
0 Schedules
01 Venues
Jul 23, 2022 - Jul 23, 2022
SAHMRI Bright Walk
21km Walk at night to raise funds for research
0 Responded
1 Guides
0 Schedules
01 Venues
Jul 23, 2022 - Jul 23, 2022
Special Olympics Fundraising Lunch
Hosted by Wellteam member Jehad Rasheed, this even...
0 Responded
1 Guides
0 Schedules
01 Venues
Aug 05, 2022 - Aug 05, 2022


Survey Activity
Activity will provide Jul 15, 2022 to Jul 09, 2022
Scorecard tool
List
3.0
2.5
2.0
1.5
1.0
0.5
0
2022-07-09
2022-07-10
2022-07-11
2022-07-12
2022-07-13
2022-07-14
2022-07-15
Survey Statistics
Total Survey
1
Active Survey
1
Total Response
43
Incomplete Response
1
Scorecard tool
TAKES 7-10 MINUTES. Respond to 45 statements across performance themes. Do statements sound 'Like y
Active
44 Responded
45 Questions

Chapter 4

Conclusion

The duration of my internship was 6 months long. It was a really nice program. Vivasoft helped us by providing accommodation during the internship period with 3 times meal. Vivasoft believes that office premises is the best learning platform for the interns. That is why they helped us to attended the office physically. I have attended the office physically from the first day of my internship. The office environment was really friendly. Within a very short period of time I got introduced with everyone.

As vivasoft doesn't have any traditional HR, everyone has easy access to managing board and they can share their problems with managing board. The managing board is so awesome they care for their employees.

I was also happy about the remuneration they provide. I find it very reasonable and satisfying according to the facilities we are provided. For full time internship, I was offered BDT 10,000 per month. They helped us to open bank account and paid the money on my account.

From the first day of intern a dedicated mentor was assigned to me. My mentor was so awesome and friendly. My seat was beside him. So, any time I was able to knock him though he was very busy with his work but helped immediately.

From my point of view, who are interested to learn and grow in a friendly environment, I would like to recommend Vivasoft. From different software company people joined at Vivasoft and everyone appreciated the friendly environment. So, I think it is one of the best places in Bangladesh to start internship or career as a software engineer as well as continue the career.

Bibliography

[1] https://www.vivasoftltd.com

[2] https://en.wikipedia.org/wiki/Internship

[3] https://www.jenkins.io

[4] https://docs.docker.com/

[5] https://kubernetes.io/

[6] https://www.mirantis.com/cloud-native-concepts/getting-started-with-kubernetes/what-is-kubernetes/

[7] https://www.terraform.io/

[8] http://tsung.erlang-projects.org/

[9] https://aws.amazon.com/

[10] https://helm.sh/