

實務專題計畫摘要報告 Narwhal – A PaaS Cloud Platform

專題編號：108-CSIE-S006

執行期限：107 年第 1 學期至 108 年第 1 學期

指導教授：劉建宏

專題參與人員： 105590010 陳威任
105810037 姚韋辰

一、摘要

本專題透過 Dockerize 技術，我們目標實現易於使用的 PaaS 平台，並將之命名為 Narwhal，中文譯作獨角鯨。我們致力於簡化及加速所有複雜之操作步驟，並希望能大幅簡化架設服務之流程，讓學生或是業餘開發者無需具備伺服器營運相關技術，能更專注於商業邏輯的設計與產品的優化。

關鍵詞：容器化、PaaS、容器管理、自動化部署、容器負載平衡

二、緣由與目的

有鑒於現行雲端平台所提供之服務，仍然有著門檻過高的窘境，對於學生及業餘人士而言，始終無法輕易跨越，我們也有收到許多非相關科系的學生來詢問如何架設自己的服務，我們聽到了這些聲音，因此 Narwhal 為此而生！

三、使用技術

(一) Docker Machine

Docker Machine 是一個工具，它允許你在虛擬宿主機上安裝 Docker Engine，並使用 docker-machine 命令管理這些宿主機。你可以使用 Machine 在伺服器上快速的建立像 AWS 或 Digital Ocean 這樣的

雲供應商上創建 Docker 宿主機。

(二) Docker

近年流行的虛擬化技術，能快速的部屬眾多隔離的系統與服務，方便我們提供安全且高效 PaaS 平台。

(三) Python

作為後端開發環境並使用 web 開發框架 - Flask，實作與使用者之間溝通的 API Server。

(四) Vue.js

時下熱門的 web 前端框架，使用 MVVM 架構來撰寫，並相較於其他框架有更好的效能，為我們提供高效能的使用者介面。

(五) Websocket – Socket.io

封裝完整且易於使用的 websocket 操作套件。

(六) MariaDB

使用關聯式資料庫儲存使用者、容器等資料，並有相較於非關聯式資料庫有更好的效能。

(七) Nginx

非同步的 Web 伺服器框架，用作反向代理、負載平衡和快取。

(八) Circle CI

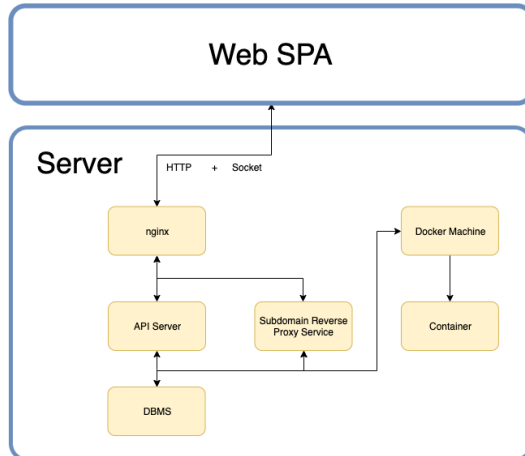
自動化整個開發流程，讓測試、分析、部署流程更快速且容易。

四、 架構流程

1. 微服務架構

將各項服務拆分成小服務，使架構更佳的靈活，也更易於維護，同時降低程式的耦合度。

2. 前後端分離架構



將前後端的拆開，使前後端部署更加的容易且靈活，也能更好的做到 MVC 甚至 MVVM 等架構出來，提升整體的可維護性，並將低耦合。

3. MVVM 架構



使用 MVVM 架構的目的是為了降低相依性，並讓程式更容易的維護及測試。且 MVVM 也改善傳統 MVC 架構上的部分問題。

五、 系統功能

1. 服務自動託管與建置

簡化伺服器建置與設定的步驟，並為使用者設定好伺服器環境。使用者不需要管理與控制伺服器，只需要專注於開發的應用程式。

2. 自動化部署

自動為使用者進行服務的

部署。

3. 客製化網址

使用者可使用客製化的網址。

4. 獲取 Docker Container

可以將現有服務與設定輸出成客製化的 Docker container 並提供下載方便遷移至其他具有 Docker 的環境。

六、 成果結論

Narwhal 利用 Docker 容器化與自動部署技術，簡化服務架設之流程，並提供資金不充裕的學生族群，有個簡單且便宜的 PaaS 服務。使用者只需將程式撰寫完成，並執行特定 shell script，即可快速將服務架設完成，甚至可以從 Narwhal 上面選用以撰寫好之服務進行快速架設。未來希望能夠提供更多服務，並使用 plugin 的形式開放開發者可自行開發靈活的 middleware 或 service，以促進平台的活躍。

七、 參考文獻

- [1] Vue.js API Documentation:
<https://cn.vuejs.org/v2/api/>
- [2] Docker Documentation:
<https://docs.docker.com/>
- [3] MariaDB Documentation:
<https://docs.mariadb.com/>
- [4] Docker Machine Documentation:
<https://docs.docker.com/machine/>