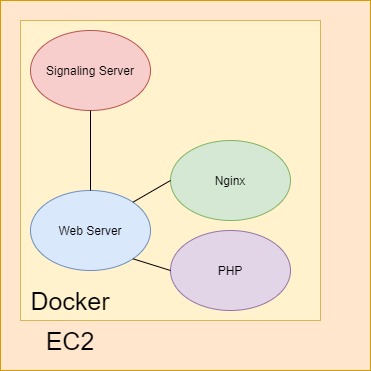
**Capstone Final Presentation**

組員:林修維、陳瑞成、柏依辰

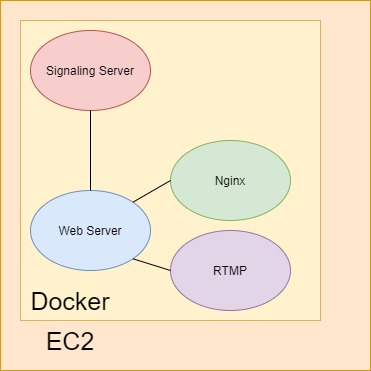
**系統架構**

PHP在第7版之後將效能以及使用體驗全面進行優化，加上Nginx本身有反向代理的功能，因此採用此架構。

**視訊: WebRTC+ Nginx+ PHP**



**直播: RTMP+ Nginx**



**設計理念**

目標的客群是初學的開發者，所以以極簡與輕鬆的風格來呈現，希望能吸引初學開發者的興趣，透過簡單的實現方式來達到輕量化以及低延遲的目的。並且希望在未來可以將我們學到的技術撰寫成像是鐵人賽之類的部落格，幫助其他也想要實現這項技術的開發者，並分享在這次系統架設過程中所學到的經驗。

**系統服務功能**

\* **視訊:** 視訊的部分是透過WebRTC原生的function，直播的部分是透過RTMP。

\* **濾鏡：** 透過WebRTC實做broadcast，讓每個直播主都有六個濾鏡可以做選擇，且在其他頁面中皆會同步顯示套用濾鏡後的結果。

\* **打賞:** 透過HTML、PHP、Javascript以及CSS來實現，透過按下可愛的按鈕來累計點數，而且不同的按鈕會有不同的點數。

\* **字幕:** 透過HTML、Javascript和CSS實作，透過跑馬燈來呈現滾動字幕。原先是想要使用VTT來製作，但比起普通的黑底字幕，跑馬燈更有生動的感覺。

\* **聊天室:** 透過HTML、Javascript和CSS實作。透過聊天與直播主互動，拉近與直播主的距離。

\* **Load Balancing:** 其中Proxy server是使用Nginx實現，不但能負載平衡，還可以預防其中的節點故障導致服務中斷所造成使用者不佳的體驗。

\* **Docker化:** 將server容器化後，可以運行在任何有docker環境的作業系統，提高了可移植性。因為以上的功能都是用很輕量的方式實現，所以容器的運行非常順暢。

**完成此課程所需的能力**

\* **Javascript:** 透過Javascript可以更方便地加入條件式，讓網頁內的物件變得更加生動。順帶一提，我們的通訊協定都是用Javascript實現的。

\* **Html:** 用簡單明瞭的格式，讓版面配置變得更簡單。

\* **PHP:** 我們聊天室和打賞的功能，都是使用PHP的語法，來實現http method POST以及GET。

\* **CSS:** 可以將object量化以及修改，做出我們要的幽默風格。

\* **協作能力：**因為要把每個人負責的東西整合在一起是很繁瑣的，因此我們學習如何使用git，而git能幫助我們整合一整個專案。

**時程規劃**

**九月：**

\* 了解串流平台的背景知識

\* 熟悉AWS的使用

\* 熟悉相關基礎的語法

\* 解決分工問題

**十月：**

\* 透過RTMP視訊

\* 透過WebRTC視訊

\* 將RTMP容器化

**十一月：**

\* 透過PHP實現聊天室

\* 透過PHP實現打賞功能

\* 將WebRTC容器化

**十二月：**

\* 字幕呈現

\* 負載平衡

\* 背景音

\* 濾鏡

**一月：**

\* 版面配置、美觀

\* Demo

**組內分工**

**林修維(通訊)：**

- RTMP

- WebRTC

- Load balancing

- Docker

- Filter

**陳瑞成(動態網頁、美觀設計)：**

- 聊天室

- 打賞

- 字幕

- 背景音

- 版面配置、CSS

**柏依辰(美觀設計)：**

- CSS

- 版面配置

- LOGO設計

- 特效

**遇到的問題與解決方案**

**九月**

- 分工不明確的問題: 上網查詢資料以補充背景知識，並分配各自有興趣的項目。

- RTMP轉檔問題: 使用hls的function使瀏覽器能夠支援播放。

- Config檔格式錯誤: 使用command line測試以及debug。

- 影像無法讀取: 設立SSL key以及CSP來提高瀏覽器的權限。

**十月**

- 無法透過nginx和signaling server建立socket連線: 先用瀏覽器訪問signaling server，讓瀏覽器拿到signaling server的SSL Key。

- 無法自動啟動容器: 使用supervisor實現自動啟動，此外還可以控制執行緒的數量。

- 聊天室的權限問題: 必須先提高檔案的權限才能夠實現HTTP Method。

- 聊天室的語法相容問題: 因為電腦跟手機的瀏覽器容錯能力是不一樣的，所以必須盡量排除所有的ERROR和WARNING。

**十一月**

- git在file system故障: 重新啟動虛擬機，或是直接手動撰寫config file。

- 打賞系統的polling問題: 使用PHP原生的function來refresh。

- PHP的啟動: 發現跟啟動nginx的方式一樣，原理是相通的，使我更清楚一系列的操作。

**十二月**

- 跑馬燈難以控制: 使用jQuery函式庫，裡面有很多方便的function可以使用，更能有效的控制CSS。

- 濾鏡的實現方式: 透過signaling server來broadcast，讓每個直播主都有數個濾鏡可以做選擇，且在其他頁面中皆會同步顯示套用濾鏡後的結果。

- 畫面重疊問題: 利用CSS將不同物件放置在不同層，接連又學習到CSS放置的位置會決定讀取的優先權。

- 引用的圖片空白: 圖片直接下載下來是相對保險的。

**一月**

* 解決瀏覽器相容性問題: 匯入參數，使所有瀏覽器都支援”navigate.getusermedia()”。

**透過分析問題與解決問題而學習到的技術**

**視訊**

- 如何讓瀏覽器拿到SSL的key

- 如何轉檔使得瀏覽器能夠播放

- 如何使用廣播實作濾鏡

- 基礎的html、JavaScript、CSS的語法

**Server**

- 學到Nginx的load balancing各種模式的演算法

- 學會如何撰寫、測試configuration file

- 設定額外安裝的套件，熟悉了許多Linux的指令

**Docker**

- 如何啟動容器

- 如何撰寫Dockerfile

- docker的網路知識

**動態網頁**

- 我們透過JS的.currentTime 來儲存每句歌詞播放的進度條，並用陣列來儲存背景音目前播放的進度，搭配跑馬燈實現卡拉OK式的滾動歌詞效果。

- 使用polling的方式來實現HTTP Method，已達到系統同步的目的。

- 學習監聽事件以及各種語法的屬性，讓網頁變得更加生動。

**透過此課程學到的內容**

- **溝通能力:** 因為這一堂是從零開始學習，所以要如何妥善分工、各取所長來達到最大化的效率，並且能夠相容使所有Component都能組合在一起，是首要必須克服的難題，因此我們在架構的選擇上花了不少時間討論。此外，透過小組進度報告，檢核自己組別這兩周的進度，除了可以練習如何在有限的時間之內表達自己的作品，也練習當一個好的聆聽者，跟其他組交流。

- **實作能力:** 因為大部分的時間都在查詢資料，實作方面我認為提升最多的是找尋資料和看官方文件的能力。未來的工作不一定是之前學過的，所以找資料的能力是必須提升的，而這一堂課讓我們提升了不少這一方面的能力。

- **資料整理能力:** 由於找到的資料參差不齊，所以過濾以及彙整資料也是一大挑戰。因此我們使用HackMD彙整各組員的想法以及資訊。

- **時間規劃:** 因為有人要實習、有人要做專題，大家都很忙碌，所以如何抽空完成這一個專題最重要的是時間控管。因為這一堂課我培養了自律的生活，同時也大幅提升我學習其他科目的效率。

- **運用數學、科學及工程知識的能力:** 這一堂課主要著重在程式語言的專業，正好可以運用我們之前修過的課程知識、以及程式語言邏輯。

- **理解及應用專業倫理，認知社會責任:** 在這次的直播中，我們一再地遇到有關瀏覽器權限以及系統檔案等問題。在調整權限的過程中也讓我們了解到，提取使用者鏡頭畫面等問題會遇到的資安問題，因此在未來我們如果能架設安全網路的話，除了訊息加密以外，也有許多資安的議題須好好維護。

- **具有良好的外語能力:** 查找資料時，真正有用的資料總是出現在國外的文獻中。因此，為了學習到大量前人所留下的智慧，我們需要大量查找國外論壇資料，也因此增進了我們的外語能力。

- **了解及尊重多元文化、時事議題與工程技術對環境、社會及全球之影響:** 原本想使用歷史較為悠久的RTMP，並搭配Nginx實現直播平台，但完成此架構後我們並不滿足現況，繼續研究延遲更低的WebRTC，這讓我們了解不同的協定之間的優缺點，並進行比較。

**- 學習提供商業平台:** 一改消費者的角度，這次站在廠商的角度提供應用程式的服務，學習如何了解以及創造符合大眾需求的功能。