**Final Documents**

B11102015余秉修/ B11102213 曾國竣/ B11102112 李家睿

**System Architecture :**

1. **HTML :** 建立網頁的標準標記語言。HTML是一種基礎技術，與CSS、

JavaScript一起用於設計網頁，與使用者介面。

1. **CSS :** 將頁面版面設計精美，美化網頁。
2. **Javascript :** 可以用來操作HTML文件的內容和結構，動態地更新頁面而無需重新載入。 它可以回應使用者的操作，例如點擊按鈕、輸入文字等，從而改變頁面的外觀和行為。
3. **Node.js :** 是一個基於 JavaScript 的後端執行環境，使用 Chrome V8 JavaScript 引擎運行代碼。專注於提供非阻塞（non-blocking）和事件驅動（event-driven）的架構，適合構建高效能、可擴展的網路應用程式。Node.js 是現代 Web 開發的重要技術之一，尤其在實時應用和 API 開發中廣泛使用。
4. **Express :** Express.js 是一個基於 Node.js 的快速、靈活且極簡的 Web 應用框架，主要用於構建 Web 應用和 API，提供了強大的工具集來處理路由、請求、回應和中間件。
5. **MySQL :** 是一種開源(免費)的 數據庫、資料庫管理系統，廣泛應用在中小型的網站中，用來配合如PHP、ASP或ASP.NET等網頁程式語言，儲存大量數據，若網站擁有後端管理程式系統(網站後台)，多須配合資料庫功能。資料庫是用來放置大量資料與檔案的一個倉庫，SQL是跟網站倉庫溝通的管員，而MySQL是用來管理倉庫的系統。
6. **WebRTC :** WebRTC 是一個 Open source project，主要在讓網頁瀏覽器和手機應用程式能夠透過簡單的 JavaScript API 進行即時的聲音影像通訊和數據傳輸。
7. **Azure :** 用戶可快速部署並管理網站環境，支援多種開發框架和工具，適用於測試、部署及擴充 Web 應用程式。透過整合 Azure 服務，開發者能輕鬆實現高效能、安全性及可擴展性的網站解決方案。
8. **Socket :** Socket 為網站開發提供即時通訊的基礎，允許伺服器與客戶端之間建立持久的連接。它支援雙向數據傳輸，適用於聊天室、即時通知及多人互動應用程式的開發。透過高效、低延遲的通訊方式，Socket 為現代 Web 應用程式實現即時功能提供強大支援。
9. **TURN Server：**TURN伺服器是 WebRTC 技術中關鍵的一部分，專門用於處理P2P連接中因防火牆或 NAT配置而無法建立直接連線的情況。TURN 伺服器的主要功能是作為中繼伺服器，允許數據通過伺服器進行傳遞，從而實現設備之間的通訊。

**System Service and Feature :**

1. Streaming platform main page
2. streamer dashboard
3. streamer chat box
4. EMOJI
5. streaming screen

**Professional Competencies required for the course :**

1. 查詢資料的能力
2. 找出答案並解決答案的能力
3. 領導團隊、分工合作
4. 良好的溝通能力
5. 普通的網頁設計能力HTML、CSS
6. 程式語言Javascript

**Design Concept :**

我們組員大部分都喜歡看Twitch平台的實況主直播，也對Twitch大部分的介面與功能有所了解，我們想嘗試以Twitch平台為模型，然後對其進行簡化和更改，讓直播平台有我們自己的特色。

**Result and analysis :**

在本次專案中，我們成功建立了一個基於 Web 技術的直播平台，並實現了核心功能如主頁、實況主儀表板、聊天室和即時串流畫面等。整個開發過程中，我們克服了多項技術難題並累積了寶貴的經驗。

伺服器部分採用了 Node.js 搭配 Express 框架進行開發。這樣的架構不僅提升了系統的靈活性與穩定性，還加速了開發流程。在安全性方面，為了保護用戶的密碼和數據，我們使用 bcrypt 進行密碼加密存取，確保即使資料庫遭到入侵，密碼仍無法被逆向破解。此外，我們引入了基於 Session 的用戶身份驗證機制，透過唯一的 Session ID 管理用戶登錄狀態，進一步提高了系統的安全性。

在直播串流部分，原本計畫採用 RTMP 協議來進行即時串流，但由於其固有的至少 10 秒延遲問題，我們最終選擇改用 WebRTC 技術。WebRTC 是一種點對點的即時通訊技術，能有效降低延遲並提升用戶體驗。同時，我們掌握了其相關技術如 ICE、STUN 和 TURN 等核心元件，成功實現了穩定的即時通訊功能。

團隊成員在專案中分工明確，各自負責不同模組的設計與開發。我們不僅學習並掌握了 HTML、CSS 和 JavaScript 的網頁前端技術，還在後端開發中熟悉了 MySQL 資料庫的使用，能夠高效進行數據的查詢與管理。我們在雲端部署方面使用了 Azure 服務，學會了如何建立虛擬機器並配置應用環境，完成了專案的線上部署。

透過此次專案，我們不僅成功打造了一個具有基礎功能的直播平台，還從中學習到多種現代化的開發技術，例如雲端服務、即時串流協議和資料庫設計等。這些技術的應用不僅解決了實際的開發挑戰，也為我們未來參與更複雜的專案提供了豐富的經驗。

**Schedule :**

**一張含有 文字, 螢幕擷取畫面, 行, 數字 的圖片

自動產生的描述**

**Job partition :**

**一張含有 文字, 螢幕擷取畫面, 圖表, 字型 的圖片

自動產生的描述**

**Encountered problems and how to solve it :**

1. 如何創建一個伺服器？解決方案：JavaScript 有內置的http模塊，而express框架是基於 http模塊封裝出來的，使用更方便，功能更強

大。

2. 為了保證密碼的安全性，需要對密碼進行加密存取。解決方案：使用 bcrypt.hash可以對密碼進行加密，加密過後密碼不會被逆向破解。

3. 對用戶進行身份驗證。

解決方案：使用Session來進行身份驗證，伺服器會在使用者登入時產生一個唯一的 Session ID，並把它傳送到客戶端。

4. RTMP會產生至少10秒以上的時間延遲。解決方案:使用後來課堂上所教的WebRTC來進行直播串流。

**Techniques you learned :**

1.Cloud : 1. 提供隨需的虛擬伺服器、儲存和網路服務

2. 提供開發和部署應用程式的工具，無需管理基礎設施

3. 透過網路提供即用型軟體

2.Create a virtual machine :

1. 進入 Azure 服務，點選「建立資源」
2. 選擇「虛擬機器」，設定名稱、區域和規格
3. 確認設定，點選「檢閱並建立」，然後「建立」

3.HTML,CSS,JS :

HTML（超內文標記語言）是用來建立網頁結構的，讓內容能夠有層次地呈現；CSS（層疊樣式表）則負責美化網頁，讓設計更具吸引力和一致性；JavaScript 是一種強大的程式語言，可以讓網頁變得更具互動性，實現各種動態功能。

4.DB :

資料庫是一種用於儲存和管理資料的系統工具，能夠高效地處理大規模資料。我們熟悉 SQL（結構化查詢語言），可以通過查詢、插入、更新和刪除操作來管理資料。我們了解資料庫的結構設計，包括表、資料欄、主鍵、外鍵等概念，並熟悉關聯式資料庫（如 MySQL、PostgreSQL）和非關聯式資料庫（如 MongoDB）的使用。

5.RTMP :

我們已經學習並掌握了 RTMP（Real-Time Messaging Protocol，實時訊息傳輸協定）的基本原理和應用。RTMP 是一種設計用於實時串流媒體傳輸的協議，廣泛用於直播和影片點播等場景。我們了解 RTMP 的資料傳輸機制，包括如何使用它來傳輸音訊、影片以及後設資料。

6.WebRTC :

WebRTC 是一種開放標準技術，能夠在瀏覽器或行動裝置之間建立點對點的音訊、影片和資料傳輸，無需額外外掛。我們了解 WebRTC 的核心元件，包括 ICE（Interactive Connectivity Establishment）、STUN（Session Traversal Utilities for NAT） 和 TURN（Traversal Using Relays around NAT），以及它們如何協助進行連接穿透。