

即時競標與限時搶購系統

Real-time Bidding & Flash Sale System

專案背景

某電商平台希望在「雙 11」或「黑色星期五」期間舉辦一場高強度的促銷活動。為了吸引流量，平台將不定時推出「限量稀有商品」（如限量球鞋、顯示卡、演唱會門票...），並採用「動態競標」與「限時秒殺」混合的模式。你們的任務是設計一個能夠承受瞬間巨大流量（Thundering Herd problem）的雲端後端系統。

競標與秒殺規則

在指定時間內（例如 10 分鐘），系統會開放一件限量商品供全體會員競標搶購。為了增加遊戲性與技術挑戰，得標者的決定不僅僅是看「出價高低」，還包含「下單速度」與「會員貢獻度」。

核心參數

- 出價金額 P (Price)：使用者願意支付的金額（必須高於底價）。
- 反應時間 T (Time)：商品開賣後，使用者送出請求的時間差（越短越好，單位為毫秒）。
- 會員權重 W (Weight)：會員的等級或歷史貢獻度。

排名/得標公式

系統需即時計算每位參與者的「搶購積分 Score」，公式如下：

$$Score = \alpha \cdot P + \frac{\beta}{T + 1} + \gamma \cdot W$$

其中 α, β, γ 為伺服器端可動態調整的權重參數。上述公式意味著參與者的 Score 由出價金額 P 、反應時間 T 、及會員權重 W 來決定。

庫存限制：若商品限量 K 個，則只有積分排名前 K 位的用戶能成功「暫定得標」。

動態競爭機制：為了模擬真實世界的激烈競爭，使用者可以在截止時間內無限次修改出價（Update Bid），試圖提高自己的排名，但這將導致後端面臨大量的 Write Requests。

系統需求

你們是該電商的架構團隊，需實作以下模組與資料庫：

- 會員系統：註冊、登入、模擬會員權重 W 的分配（可由系統隨機分配或預設）。

- 活動後台：管理員可上架商品、設定庫存量 K 、底價、與參數 α, β, γ 。
- 競標介面：使用者查看商品、送出出價、更新出價。
- 即時戰況看板：動態公布目前「前 K 名暫定得標者」及其積分（類似排行榜），以及目前的「最高出價」與「最低得標門檻分數」。
 - ◆ 技術難點：此看板需接近即時 (Real-time) 更新，且在高並發 (concurrency) 下不能顯示過舊的資料。

技術規範

- 必須自行設計測試程式與資料，進行壓力測試。
- 介面僅需陽春呈現重要資訊（登入、出價、即時排名），介面美觀非評分標準。
- 系統需用 Container 包裝服務元件，並佈署到 AWS 或 GCP。
- 資料一致性要求：必須確保絕對不能發生「超賣」(Over-selling) 的情況。即庫存只有 5 個時，最終成交訂單絕不能超過 5 筆。

期末 Demo 簡報內容 (10~15 pages)

- 系統簡介與架構圖。
- 核心技術說明
 1. 如何處理高並發的寫入（大量的出價更新）？
 2. 如何計算（含參數值 P, T, W, K 與 α, β, γ 設定）並推播即時排名？
 3. 如何保證庫存一致性
- 使用的平台、工具、套件
- Scalability 設計：當流量暴增時，系統如何自動擴展？
- 系統測試設計與壓力測試數據
- 系統的容錯能力 (optional)
- 分工架構
- Demo 影片：影片長度不超過 3 分鐘，必須展現（請搭配語音或字幕說明）：
 1. 系統啟動、商品上架設定。
 2. 使用者操作：登入、出價、即時看到排名變化。
 3. 模擬搶購過程（壓力測試）：模擬至少 1000 個 concurrent users 同時對同一商品進行競標與搶購。
 4. 隨著截止時間接近，更新出價的頻率須呈現指數型成長。

5. Scalability 展示：展示 CPU 使用率上升時，Container/Instance 數量的變化，以及 Response Time 的穩定性。
6. 一致性驗證：在大量並發結束後，展示資料庫結果，證明沒有超賣（成交數 \leq 庫存數）。

繳交日期

- 12/10: Midnight: 上傳簡報與影片
- 12/11: Term Projects Demo
- 12/12: 打包 (zipped) 你們所有的程式碼及說明文件並上傳到 NTU cool