**NP.edu**

1083315 吳承軒

1083321 葉品和

1. **提案構想**

本app將會詳細介紹所有問題的分類，包括 P-problem、NP-problem (NP-complete、NP-hard)、intractable-problem、unsolvable-problem，使用者可以點選對應選項以查看問題的說明，也能點選對應的演算法查看介紹及分類，最後會有一個小測驗來當作使用者吸收程度的參考。

1. **創新特色**

* 整合並分類多種演算法 (現有的資訊大多是雜亂的，我們將之統整在一起)。
* 清楚的使用者介面讓使用者能快速上手 (無須經過教學就能完整使用app)。
* 讓使用者能快速學習NP-problem的知識。
* 不只是單純的閱讀資訊，而是加上測驗使記憶更深刻。

1. **相關作品**
2. 知識王：與app最後的小測驗有些相似，同樣是問答，我們的特色是做完測驗後會完整解答題目。
3. Quizlet：與app前半部較為相似，能通過整合的資訊學習，而我們的app是專門針對NP-problem的。
4. **系統設計**
5. 系統軟硬體設計：API 30。
6. 系統架構設計：布局、文字方塊、按鈕、換頁。
7. 功能性要求：以測驗方式與使用者互動。
8. **介面設計**

|  |  |
| --- | --- |
| 圖一、主畫面 | 圖二、選擇problem畫面 |

|  |  |
| --- | --- |
| 圖三、選擇問題簡介或演算法畫面 | 圖四、問題簡介畫面\*5 |

|  |  |
| --- | --- |
| 圖五、演算法選擇畫面\*5 | 圖六、演算法介紹畫面\*n |

|  |  |
| --- | --- |
| 圖七、測驗畫面\*10 | 圖八、測驗結束畫面 |

|  |
| --- |
| 圖九、檢討畫面 |

1. **未來工作**
2. 時程規劃：期中考後固定每天晚上討論一小時，假日視情況增減。
3. 工作項目規劃：首先收集問題及演算法的資料，整理完畢後開始進行app設計，首先是設計介紹問題畫面，接著是設計介紹演算法畫面，最後完成小測驗。
4. 分工實作方式：報告部分採用google雲端文件，以語音的方式同步進行討論與編寫，app部分一人先拉layout，另一人寫code，互相合作。
5. **參考資料**
6. https://en.wikipedia.org/wiki/List\_of\_unsolved\_problems\_in\_computer\_science
7. https://bluelove1968.pixnet.net/blog/post/222283186
8. https://www.jntua.ac.in/gate-online-classes/registration/downloads/material/a159262902029.pdf