

六、應用實例

在本章中將會展示以視覺化軟體建構方法來開發行動電話使用者介面，首先由使用視覺化需求編輯系統開始，再搭配樣板，到使用功能連結系統，最後產生程式碼在模擬器上執行。下圖為應用實例的流程，區分為在第3章所述開發流程的三個階段，視覺化需求編輯階段，程式碼產生階段，系統模擬階段，將由Start開始，直到End結束。

1. 新增或載入一個專案
2. 新增或刪除一個場景
3. 除了自行建立使用者介面，也可使用樣板管理器，建立完整的使用者介面
4. 建立完整的使用者介面後，接下來可編輯使用者介面的內容
5. 完成使用者介面的編輯後，接下來做功能連結，建立使用者介面與功能上的關係。
6. 完成功能連結後，啟動code generator將編輯的結果轉換為目標平台上的程式碼。
7. 最後將產生出來的程式碼與模擬器編譯執行。

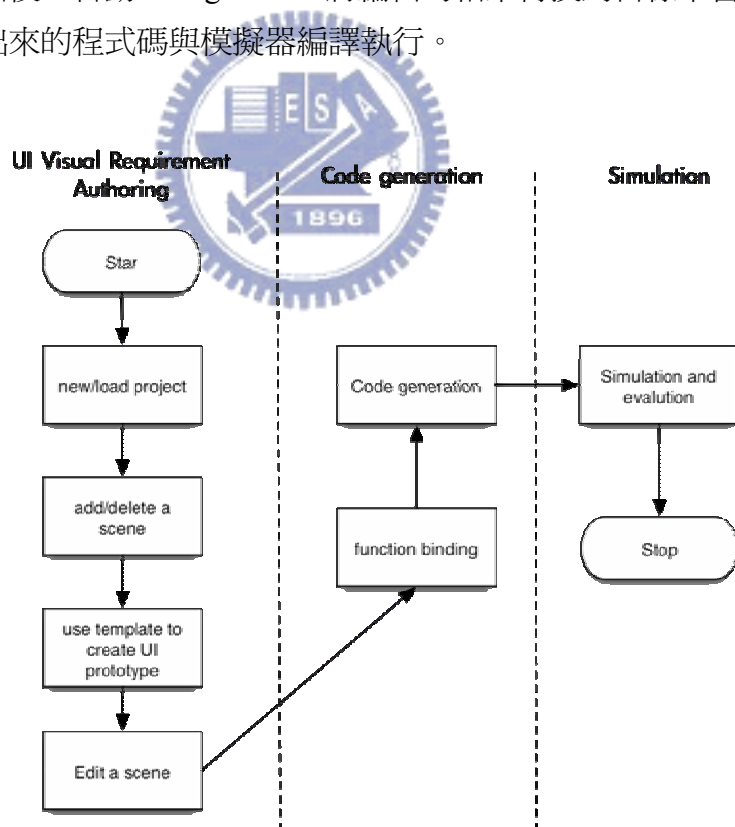


圖 29 行動電話使用者介面開發展示流程圖

6.1 視覺化的使用者介面需求編輯階段

當我們要開始要製作使用者介面時，首先會進入 專案管理畫面，如圖 30，在此畫面可以選擇要新增一個專案還是載入之前所編輯的結果。

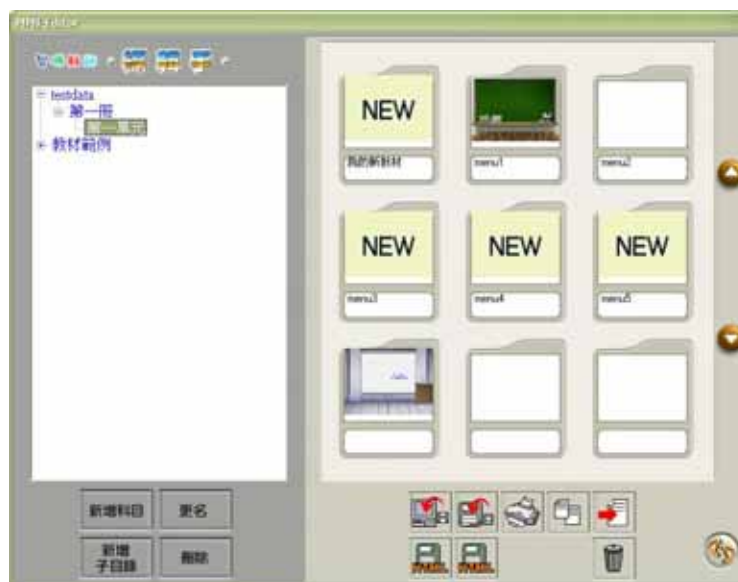


圖 30 專案管理畫面

當在專案管理畫面的選擇結束後，我們進入了場景的管理畫面，如圖 31，此為系統的主控制台，提供建立/刪除場景的結構，提供樣板，功能模組設定，產生目標平台程式碼功能。

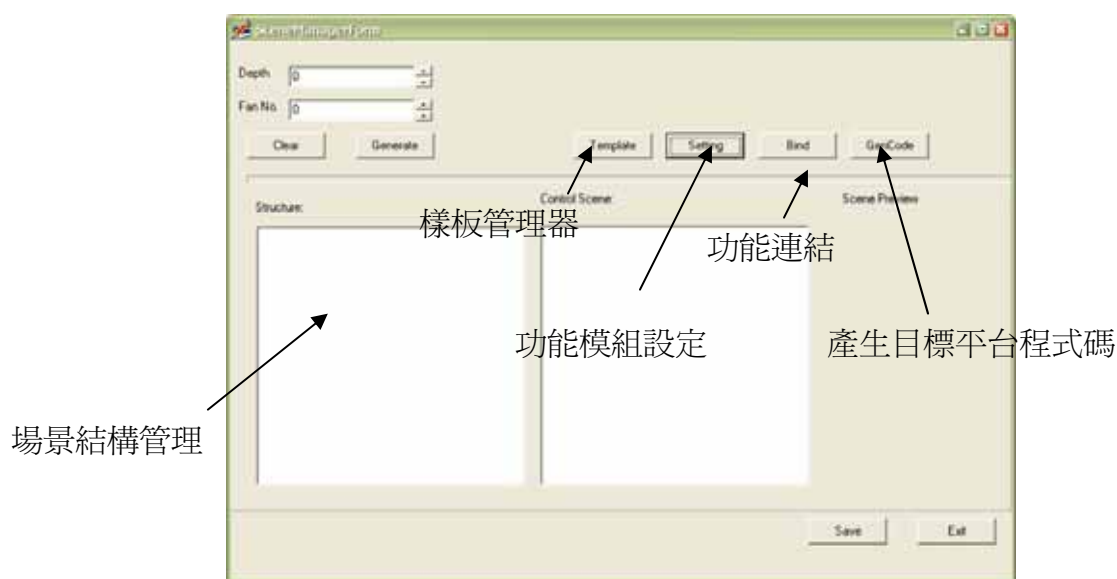


圖 31 場景管理畫面

如果在專案管理畫面中，選擇載入舊有的編輯結果，則會在場景結構管理部份出現舊有的編輯結果，如圖 32 左半部所示。

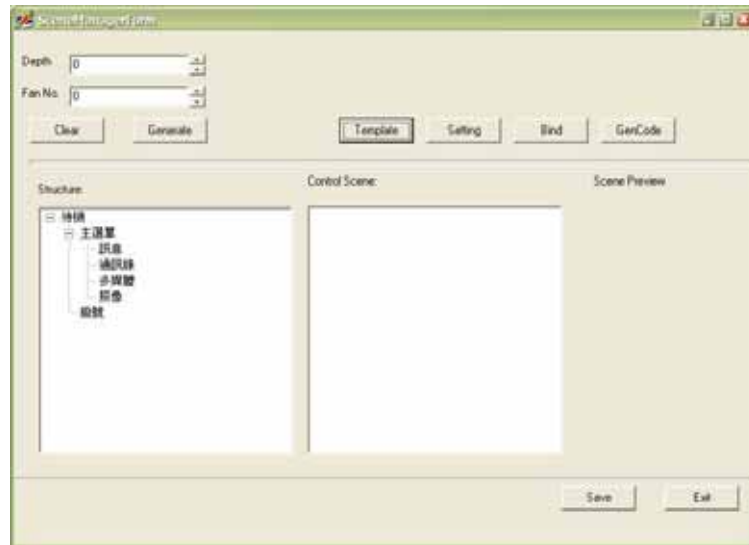


圖 32 舊有的編輯結果

若在專案管理畫面上，選擇新增一個專案，則可以使用樣板管理器來協助建立使用者介面或重新開始建立，或由自行重新建立場景。

若使用樣板管理器，它提供了三種樣板，結構樣板(Structure template)、版面樣板(Layout template)、樣式樣板(Style template)，如圖 33, 34, 35。

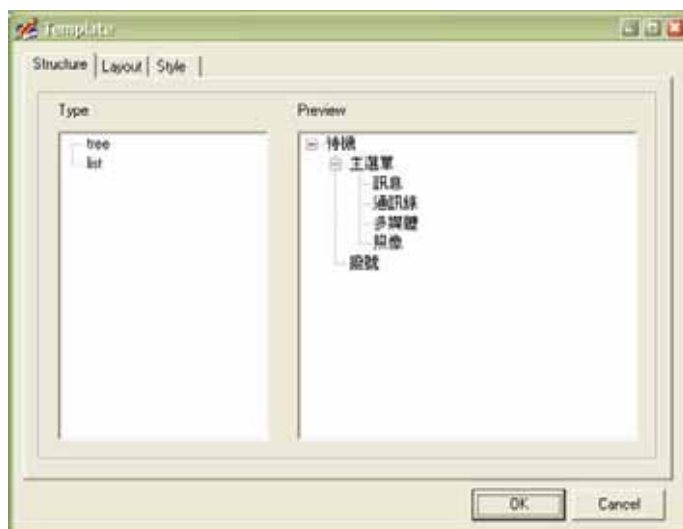


圖 33 結構樣板

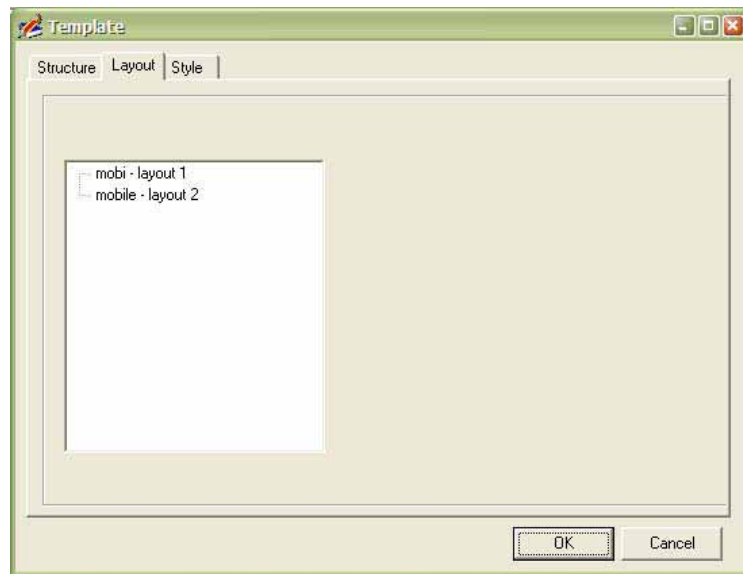


圖 34 版面樣板

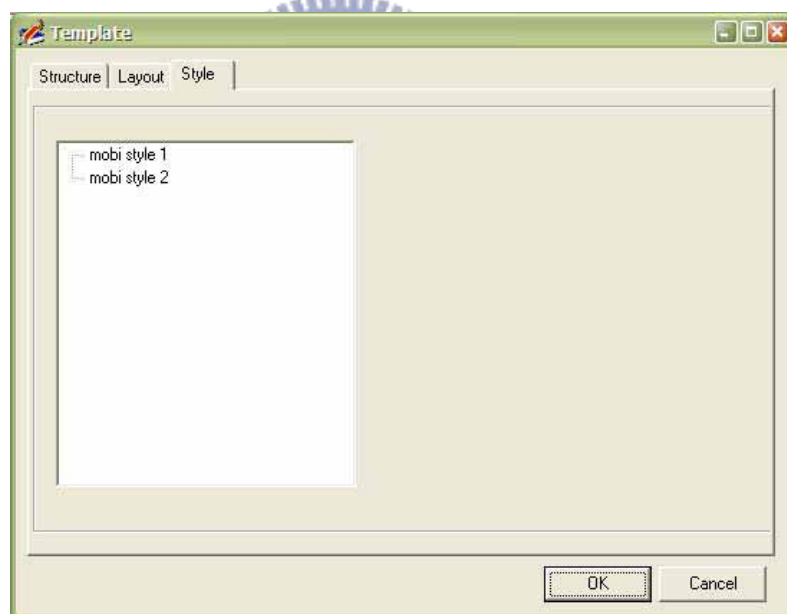


圖 35 樣式樣板

我們先來看不同樣板搭配下所帶來的影響：

版面樣板不同的情形

文字在下的版面樣板，如圖 36。



圖 36 文字在下的版面

文字在上的樣板，如圖 37。

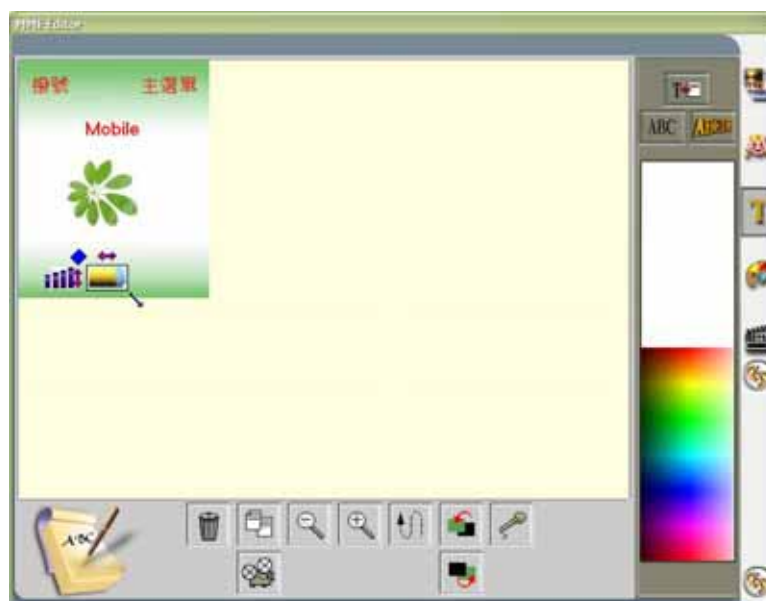


圖 37 文字在上的版面

樣式樣板不同的情形

以文字為主的樣式樣板，如圖 38。



圖 38 以文字為主的樣式

以圖示為主的樣式樣板，如圖 39。

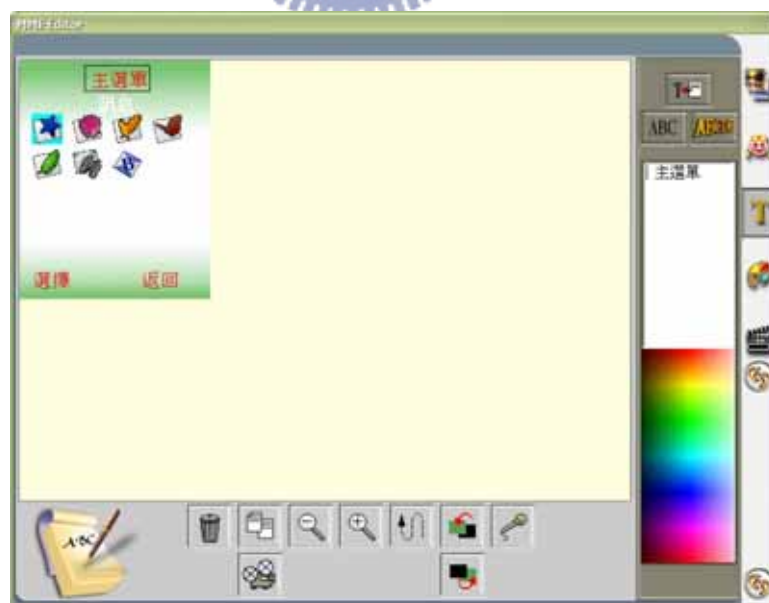


圖 39 以圖示為主的樣式

在使用樣板管理器後，我們得到了使用者界面的雛型，如圖 40。

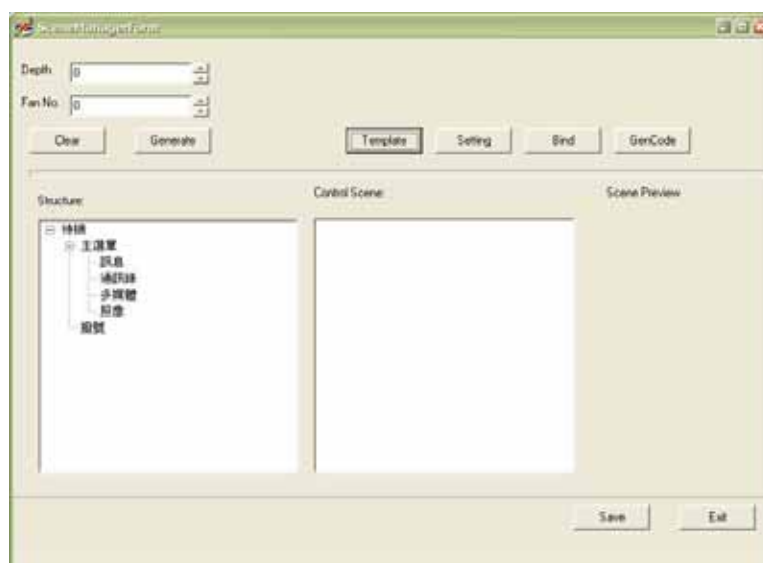


圖 40 由樣板管理器產生使用者界面雛形

當完成場景的新增後，我們可對各個場景做進一步的編輯，接下來進入使用者界面需求視覺化編輯系統內，做畫面的編輯，我們可以新增一個新的文字或圖片或做場景與場景之間的連結。如圖 41

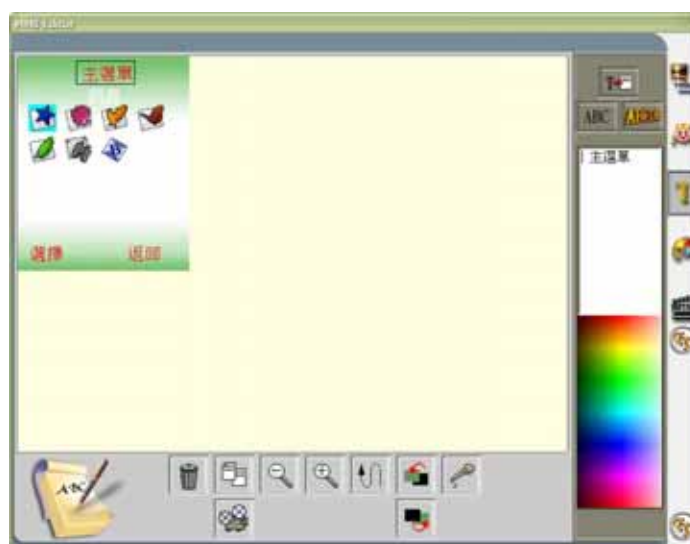


圖 41 編輯系統畫面

在編輯的過程中也可以試撥來確認實際運作的情形，可以以鍵盤來模擬操作的情形，在場景與場景之間跳躍。圖 42 為撥放的情形。



圖 42 試撥畫面



6.2 程式碼產生階段

在完成使用者界面的編輯後，接下來將目標使用者介面與基本功能或附加功能連結，我們回到場景管理畫面，先選擇要使用的功能模組，來決定接下來所能連結的功能種類，如圖 43 所示。

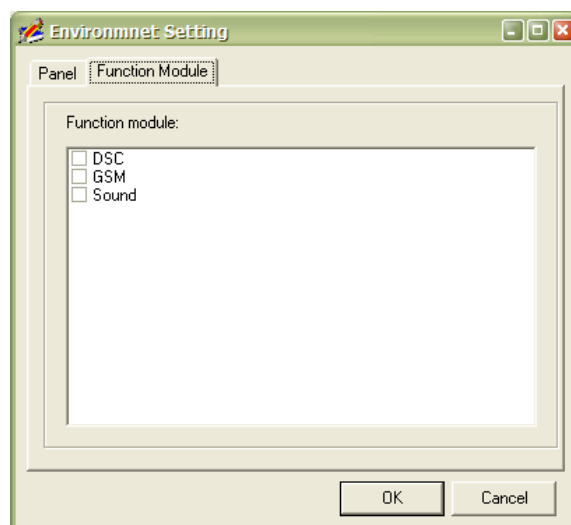


圖 43 功能模組選擇

因為在模擬的環境只提供相機模組及 GSM 模組，因此選擇 DSC 與 GSM，接著回到場景管理畫面，選擇要做功能連結的場景，選擇功能連結功能。

接著進入了功能連結系統，如圖 44 所示，點擇要做功能連結的演員，選擇功能連結



圖 44 功能連結系統

為演員設定觸發執行的功能，如圖 45 所示，列出了屬於 DSC 與相機模組的功能

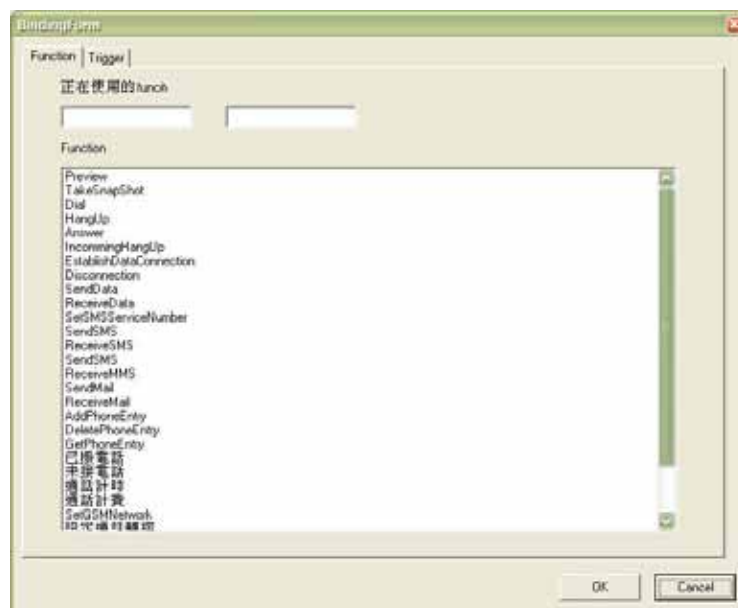


圖 45 功能清單

再為演員設定觸發功能的按鍵，如圖 46 所示

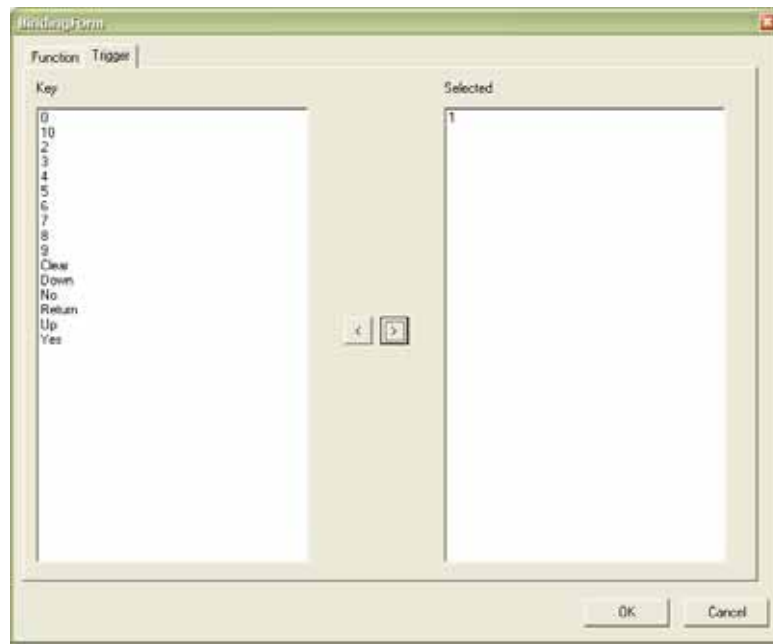


圖 46 設定觸發按鍵

當完全所有的功能連結後，則回到 UI 管理畫面，執行產生目的碼功能，產生出目標平台的程式碼，如圖 47 所示

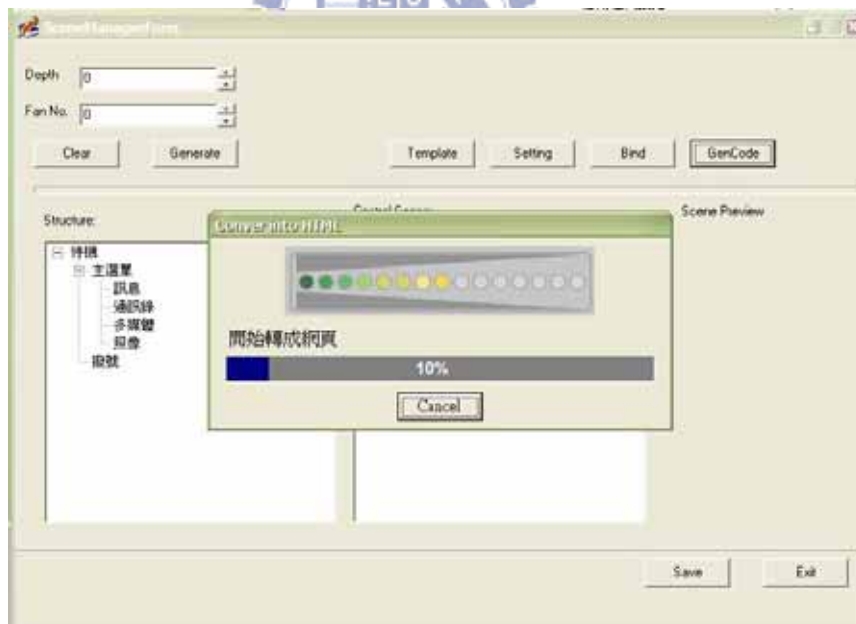


圖 47 產生目標平台程式碼

接下來將產生出的程式碼與模擬器編譯即可。

6.3 系統模擬

藉由模擬器，可以驗證 UI 設計的結果，並能執行相對應的功能，撥電話，拍照等功能，我們提供了數位相機與行動電話兩種模組，以下圖 48 為模擬的環境。

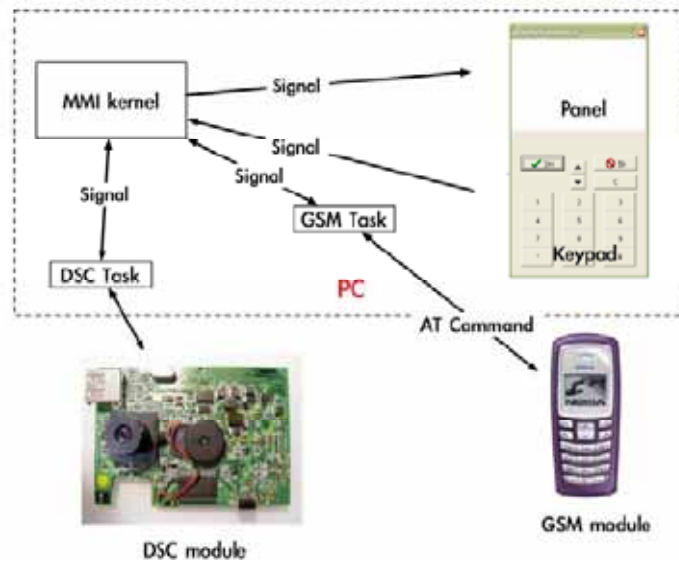


圖 48 模擬環境

以下為模擬器執行結果：

藉由模擬器可以操作選單並執行功能，如圖 49 為選單操作畫面。



圖 49 選單操作畫面

如圖 50 為撥號與拍照功能



圖 50 撥號與拍照功能



七、總結

7.1 結論

在使用者的介面開發上，一直是耗費時間的工作，程式設計師為主要的開發者。導入視覺化的軟體建構方法，可以降低程式設計師的工作負擔，他們只需專注於開發系統的功能上。而借由視覺化的編輯器協助，降低了開發使用者介面的門檻。使得開發使用者介面時，不用寫程式碼，只要以視覺化的方式，以滑鼠點選、拖拉，將各個可重複使用的演員組合成各個畫面，再把各畫面，組合成完整的使用者介面。

因此使用者介面設計者，便能利用這個工具來直接設計並且實作出所想要的使用者介面，而不必依賴程式設計師將使用者介面實作出來。

而透過程式碼產生器，則賦與使用者介面真正的功能，結合程式設計師所開發出來的各個函式庫，產生出真正的程式。我們只需在使用者介面上加上與各個功能間的連結。

藉由 GKI 所提供的介面協助，我們能將使用者介面與實際上的硬體與作業系統隔離開來，當底層做修改時，使用者介面需要的更動能達到最小。

而樣板的加入，當在開發相類似的使用者介面時，能快速的產生使用者介面雛型，再進一步的修改成客製化的使用者介面，提高開發的效率。且透過三種樣板，結構樣板、版面樣板、樣式樣板之間不同的組合，能產生相同結構不同外觀，或不同結構相同外觀的使用者介面

7.2 未來發展方向

1. 提供個人化的功能

由於個人化的風潮，部分軟體已有讓使用者自行更換使用者介面外觀的功能，此系統所開發的使用者介面已在開發階段就固定下來，在未來可再進一步延伸，能讓使

用者自己設計屬於自己的使用者介面，或在網站上提供可讓使用者自行下載，提供更強大的消費者端客製功能。

2. 提供標準使用者介面元件

在編輯環境方面，能訂定行動電話使用者介面的標準演員，使用者介面設計者使用這些演員即能完成所有的使用者介面設計，如同 Windows 上的標準使用者介面元件。



參考文獻與資料

- [1] 拓樸產業研究所, <http://www.topology.com.tw>
- [2] Ian Sommerville, Software Engineering, 6th Edition, Addison Wesley, 2001
- [3] Collings, P., Fransen, P., "Man-machine interfaces, trends and needs", Pulp and Paper Industry Technical Conference, 1995., Conference Record of 1995 Annual, Vol., Iss., 12-16 Jun 1995, Pages:1-16
- [4] Jia-Chen Dai, "Visual-Based User Interface Generator", N.C.T.U Taiwan, Master Thesis, 2002
- [5] Dong-Ying Lee, "An Authoring Environment for the Integration and Presentation of 2D and 3D Courseware Patterns", N.C.T.U Taiwan, Master Thesis, 2003
- [6] Acosta, R.D., Burns, C.L., Rzepka, W.E., Sidoran, J.L., "A case study of applying rapid prototyping techniques in the Requirements Engineering Environment", Requirements Engineering, 1994., Proceedings of the First International Conference on, Vol., Iss., 18-22 Apr 1994, Pages:66-73
- [7] 侯俊傑, 深入淺出 MFC 第二版 Dissecting MFC 2nd Edition, 初版, 松崗, 1997
- [8] Symbian OS – the mobile operating system, <http://www.symbian.com/>
- [9] Wills, C.E., "User interface design for the engineer", Electro/94 International. Conference Proceedings. Combined Volumes., Vol., Iss., 10-12 May 1994, Pages:415-419
- [10] Chorng-Shiuh Koong, "A Component-based Visual Scenario construction environment for non-programming users to create interactive electronic books", N.C.T.U Taiwan, dissertation, 2000
- [11] Landay, J.A., Kaufmann, T.R., "User Interface Issues in Mobile Computing", Workstation Operating Systems, 1993. Proceedings., Fourth Workshop on, Vol., Iss., 14-15 Oct 1993, Pages:40-47
- [12] Gamma, Helm, Johnson, Vlissides, 葉秉哲, Design Pattern, 普林帝斯霍爾, 2001
- [13] Wu-Chi Chen, "A Visual and Reuse-based Paradigm for Software Construction", N.C.T.U Taiwan, dissertation, 1998
- [14] Eric A.Meyer, Cascading Style Sheets: The Definitive Guide, O'Reilly, 2002
- [15] Chwan-Hung Wang, "On the Enhancement of an Multimedia Authoring Tool for the Visual-Based User Interface Requirement Representation", N.C.T.U Taiwan, Master Thesis, 2002
- [16] Joseph A. Goguen, "Formality and Informality in Requirements Engineering", Proceedings of the 2nd International Conference on Requirements Engineering, April 15-18, 1996, Colorado Springs, Colorado, pp.102-108
- [17] Ohnishi, A.; Tokuda, N., "Visual software requirements definition environment", Computer Software and Applications Conference, 1997. COMPSAC '97. Proceedings., The Twenty-First Annual International, Vol., Iss., 11-15 Aug 1997 ,Pages:624-629
- [18] D.J. Chen, W.C. Chen, K.M. Kavi, "Visual requirement representation", The Journal of Systems and Software 61, 2002, pp.129-143