IJRDET

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 4, October 2014)

Android App Design of Static and Dynamic Visual-Aids for House Selling Service Using App Inventor 2

Huang Li-Jeng[1], Liao Wan-Ling[2], Ha Thi Thuy-Van[2]

[1]Associate Professor, [2]Undergraduate Students, Department of Civil Engineering, National Kaohsiung University of Applied Science, 80778, Taiwan, Republic of China

***Abstract*— House selling service is an important task for architectural company or personal house sellers. App Inventor is a no-code open platform for Android mobile app development using drag-and-drop approach and visual programming environment. This paper is aimed at Android programming using App Inventor for professional application, i.e., static and dynamic visual aids for house selling service (HSS). System development procedures, layout design and program coding will be explained. A typical real house selling in Ping-tung, Taiwan will be employed to show the results of pictures and movies visual-aids for house selling service. The results are satisfactory and convenient for the use of house sellers.**

***Keywords*—Android Apps, App Inventor, House Selling Service, Smart Phones, Static and Dynamic Expression.**

I. INTRODUCTION

In general selling a personal own or many new houses of a construction project of company requires a lot of skills. The basic procedure includes (1) getting ready to sell; (2) choosing sale price and method; (3) marketing the property of house; (4) getting legal help; (5) negotiating the sale; (6) using an agent [1]. However, in the third step we should make sure the house and gardens are their best looking as well as put together a summary and quick guide with details of the house for selling [2].

The 28th principle of the 40 inventive principles in marketing, sales and advertising [3] is the mechanics substitution which includes to replace a mechanical means with a sensory means as well as to use electric fields to interact with the object or system (electronic communication, trade, and computerized interview, etc.) It is well known that good relationship between sellers and buyers in the first interview help to build up long-term buyer-seller relationship and increase the potential buyers for the products [4-7].

In Taiwan if people sell their own house, by themselves or by vendors, there are little visual aids employed. However if a group of new built houses sold by a company, some visual aids will be designed and employed for the help of people who take the responsibility of selling service. For the sale project of a group of new house the housing company may assign a professional teams for this purpose and prepare many kinds of sale-aided materials, i.e. three-dimensional expression models, visual aids of movies, pictures, photos and some decorated rooms with internal design.

The basic features of Android mobile applications which are adequate for the aids of house selling are summarized in the following: (1) data storage and retrieving; (2) multimedia support (sounds, pictures and movies); (3) multiple languages support; (4) web pages browsing; (5) bluetooth, wi-fi and WiMAX communications; (6) GPS and Map information; (7) multi-point touch; (8) multi-task processing; etc. [8].

Basically there are three kinds of development tools for mobile applications executable on Android or iOS systems: (1) Native Apps: e.g. Objective-C for iOS, JAVA or Badic4Android for Android, and C# or VB.NET for Windows Phone; (2) Web Apps: e.g. HTML5 and CSS3, ASP.NET, Visual Studio and JQuery Mobile; (3) Hybrid Apps: e.g. Web View, UIWeb View, PhoneGap, Titanium, AppMobi, etc.

App Inventor and its new version, App Inventor 2, are tools to design Apps for the Android platform. They are developed originally by Google and then maintained by MIT. Due to its ease of learning and usage the users increased dramatically and became a popular tool for app development. These two tools provide programmers very easy and friendly environment to develop Android apps. [1]

Some special features include: (1) object oriented programming (OOP) tool; (2) graphical user interface (GUI) environment; (3) drag-and-drop approach in layout design phase and code design phase instead of traditional line-by-line in text JAVA programming; (4) providing testing simulator on PC to check the results on Android platforms; (5) providing many ways in the deployment phase for constructing the executable APK files, e.g. QRcode, mailing APK to platform or transferred by USB port, etc. to be installed and set up on the Android based smart phones and pads; (6) the projects developed are stored and managed on web, programmer can retrieved from different platforms for his convenience. (7) free web development environment with tiles [9-12].

The major tools provided in App Inventor 2 are: (1) Component Designer, where interface and integration of components (both visible and non-visible components); (2) Block Editor, where program logic is created; and (3) Emulator for the situation program test can be conducted without connecting to a mobile phone. The design and implementation process allow people without programming skills and prior programming experience to rapidly develop mobile apps. Especially the drag-and-drop visual programming approach let users easily design and arrange interface components (Command button, Text button, Checkbox, etc.) and connect logic blocks to create their mobile applications.

Since pilot in late 2009 and the release of App Inventor in December 2010 developed by Google, led by Professor HaroldAbelson, and later moved to MIT (Wikipedia, 2014) [13], a variety of studies have been conducted on many fields: game design (Wolber, 2011; Wolber et al., 2011), educator training (Hsu et al., 2012), information security (Arachchilage et al., 2012), teaching and assessing programming (William, 2013), healthcare (Zhang, 2014), emergency management (HuiRu Shih, 2014) etc.[14-19]. Huang et al. (2014) has attempted to employ App Inventor 2 to design Android app for static structural analysis based on direct stiffness method [20].

This paper is aimed at application of App Inventor to design an Android app for static and dynamic visual aids for house selling service. The design concepts, from conceptual, preliminary to detailed, design viewers, programming blocks as well as results are depicted to show a complete process.

II. Development Of The System

The development of the system of Android app for static and dynamic visual expression for house selling is explained as follows:

A. Design Procedure of Android App for Visual Expression for House Selling

The design procedure of development of Android App for house selling service can be observed as shown in Fig.1.

B. Information Prepared for Visual Expression

Basic concept of the design of this app is to help the vendor to provide information for the customers who are potential buyers of the houses. The required important information includes:

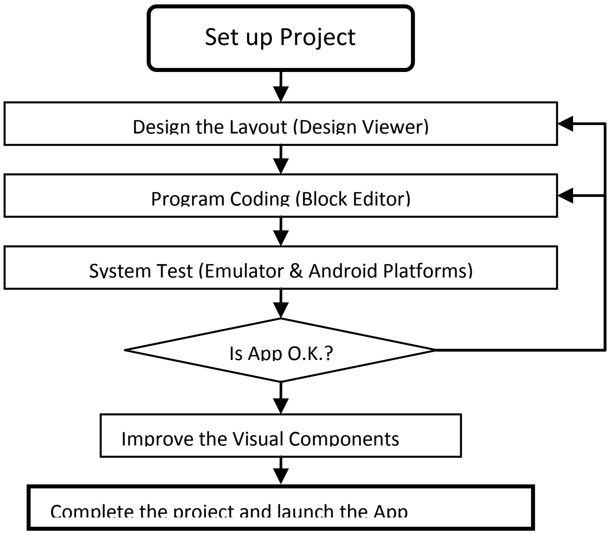

Set up Project
Design the Layout (Design Viewer)
Program Coding (Block Editor)
System Test (Emulator & Android Platforms)
Is App O.K.?
Improve the Visual Components
Complete the project and launch the App

Figure 1 Flow chart of Android App system development using App Inventor

1) Information of building (area of land, area of building, number of floor, material of construction, location, designer, constructor, date of release, etc.) ;
2) Information of house (number of rooms, toilets, kitchen, garden, etc.);
3) Information of environment;
4) Information of space design of each floor;
5) Information of structural system;
6) etc.

and finally it is convenient for the seller if there exists at least one screen to input and save some data of potential buyers, e.g., name, telephone, date of interview, and if possible the personal photo.

Furthermore, we can make decision that some information (text and/or picture) are presented in static style while some (movies) are played in dynamic mode.

The framework for designing an App for general house selling can be referred in Fig. 2. This will be employed in the use of block editor when App Inventor is conducted in programming the app.

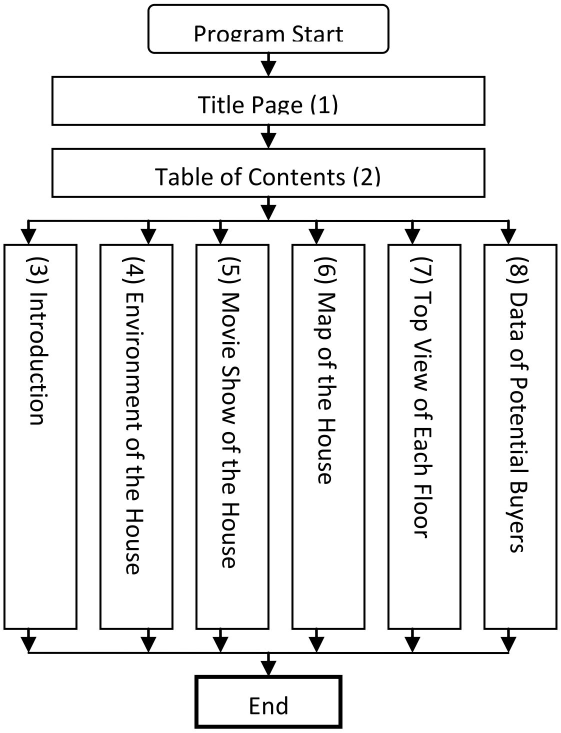

Program Start
Title Page (1)
Table of Contents (2)
(3) Introduction
(4) Environment of the House
(5) Movie Show of the House
(6) Map of the House
(7) Top View of Each Floor
(8) Data of Potential Buyers
End

Figure 2 Flow chart of Android app design of house selling service

C. App Design

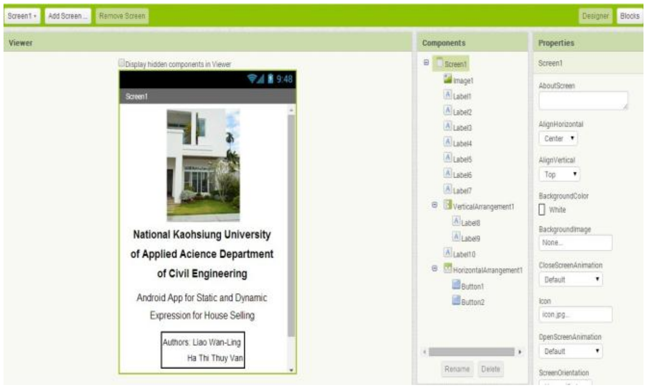
C1. Conceptual Design Phase

We design the Android app for house selling service to be one with multi-screens framework and assign it a name HSS. The conceptual design of the system can be shown in Figure A in *Appendix*. Some detailed arrangements might be changed and modified during the Preliminary Design Phase (Layout Design Phase) and rhe Detailed Design Phase (Programming Coding Phase).

C2. Layout Design Phase

We employ the multi-screen concept to design this system with 8 layouts (Screens) via *design viewer of App Inventor*. These 4 layouts (Screens) design can be observed from Fig. 3 to Fig. 10. In each screen the common used objects are command buttons, textboxes, labels etc.:

(a) Screen 1 (Homepage/Title Page):



National Kaohsiung University
of Applied Acience Department
of Civil Engineering

Android App for Static and Dynamic
Expression for House Selling

Authors: Liao Wan-Ling
Ha Thi Thuy Van

Screen1
AboutScreen
AlignHorizontal: Center
AlignVertical: Top
BackgroundColor: White
BackgroundImage: None...
CloseScreenAnimation: Default
Icon: icon.jpg...
OpenScreenAnimation: Default
ScreenOrientation

Figure 3 Screen 1(title page) of the Android app HSS

(b) Screen 2 (Table of Contents):

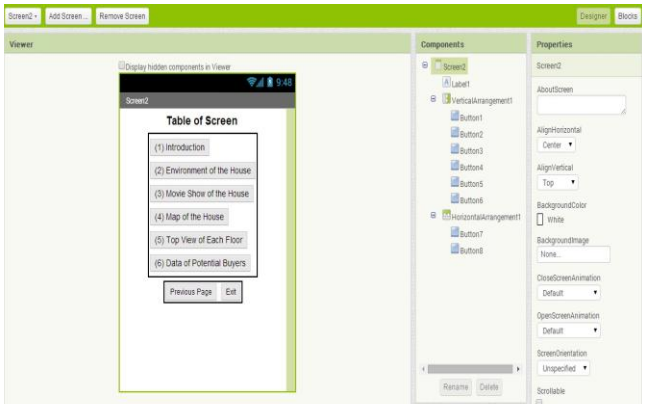


Table of Screen

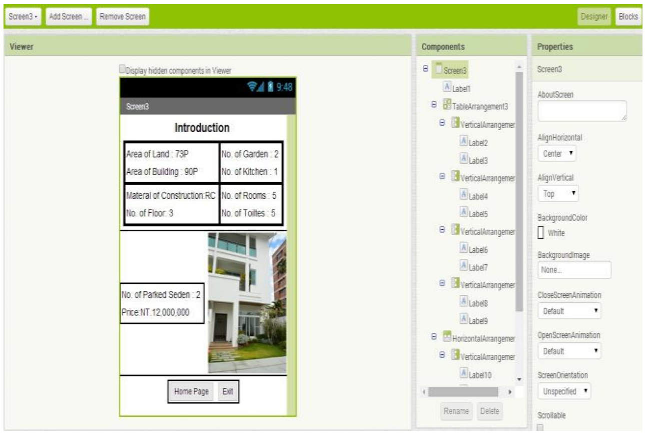
(1) Introduction
(2) Environment of the House
(3) Movie Show of the House
(4) Map of the House
(5) Top View of Each Floor
(6) Data of Potential Buyers

Previous Page | Exit

Screen2
AboutScreen
AlignHorizontal: Center
AlignVertical: Top
BackgroundColor: White
BackgroundImage: None...
CloseScreenAnimation: Default
OpenScreenAnimation: Default
ScreenOrientation: Unspecified
Scrollable ☐

Figure 4 Screen 2 (table of contents) of the Android app HSS

(c) Screen 3 (Introduction):



Introduction

Area of Land : 73P Area of Building : 90P	No. of Garden : 2 No. of Kitchen : 1
Materal of Construction:RC No. of Floor: 3	No. of Rooms : 5 No. of Toiltes : 5

No. of Parked Seden : 2
Price:NT.12,000,000

Home Page | Exit

Screen3
AboutScreen
AlignHorizontal: Center
AlignVertical: Top
BackgroundColor: White
BackgroundImage: None...
CloseScreenAnimation: Default
OpenScreenAnimation: Default
ScreenOrientation: Unspecified
Scrollable ☐

Figure 5 Screen 3 (Introduction) of the Android app HSS

(d) Screen 4 (Environment of the House):

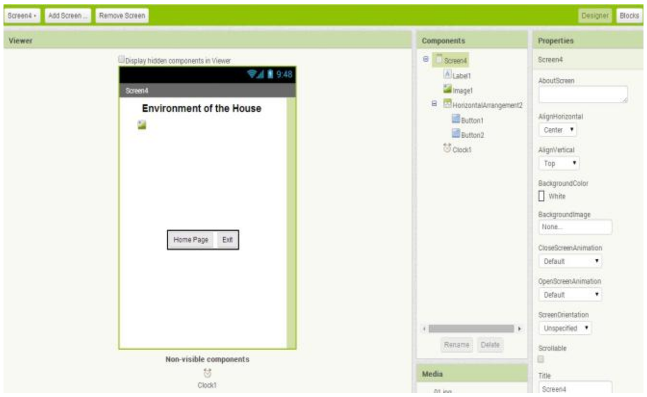


Figure 6 Screen 4 (Environment of the House) of the Android app HSS

(e) Screen 5 (Movie Show of the House):

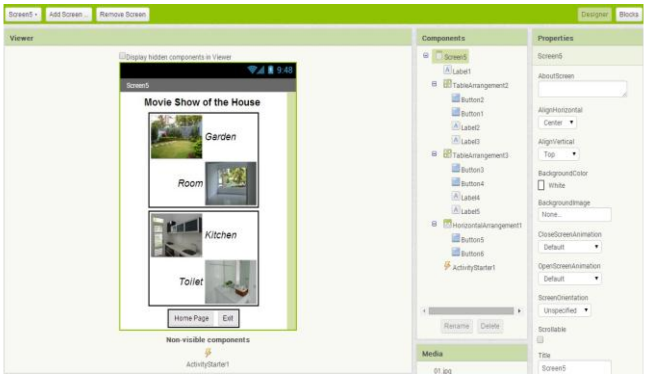


Figure 7 Screen 5 (Movie Show of the House) of the Android app HSS

(f) Screen 6 (Map of the House):

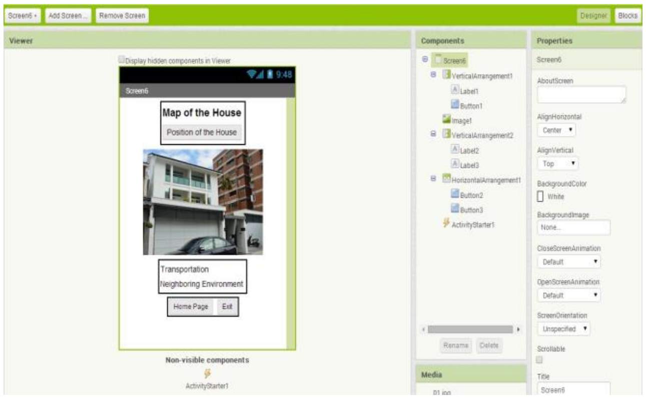


Figure 8 Screen 6 (Map of the House) of the Android app HSS

(g) Screen 7 (Top View of Each Floor):

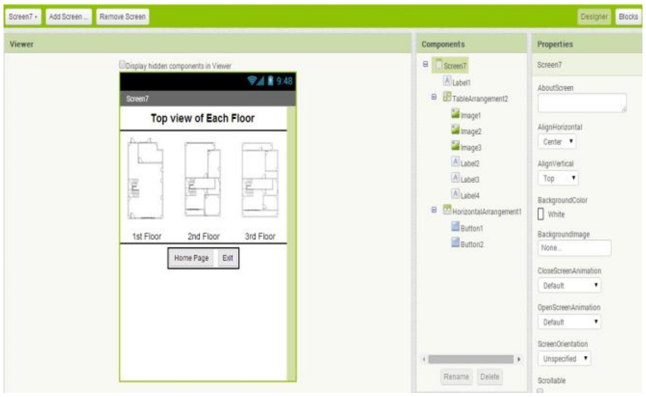


Figure 9 Screen 7 (Top View of Each Floor) of the Android app HSS

(h) Screen 8 (Data of Potential Buyers):

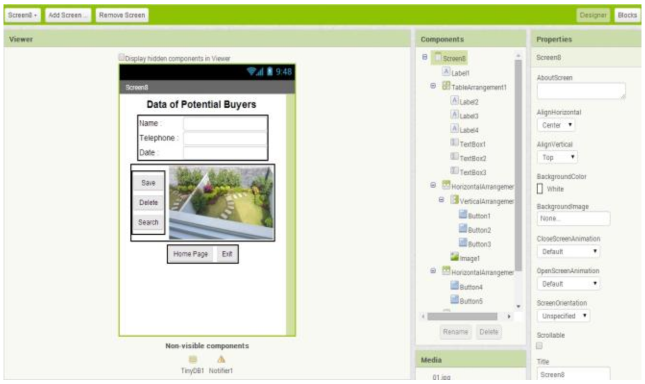


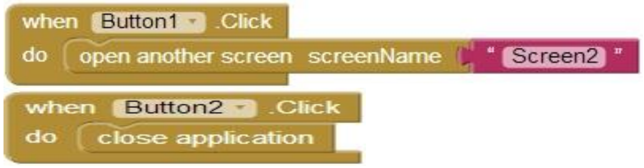
Figure 10 Screen 8 (Data of Potential Buyers) of the Android app HSS

C3. Program Coding Phase

One of special feature of App Inventor is the programming mode the same as building blocks as shown from Fig. 11 to Fig. 18.

(a) Screen 1 (Homepage/Title Page):

In this screen only two command buttons involved, one is to move to screen 2 (table of contents) and the other close application (exit of app). User also can use hard button to close application.

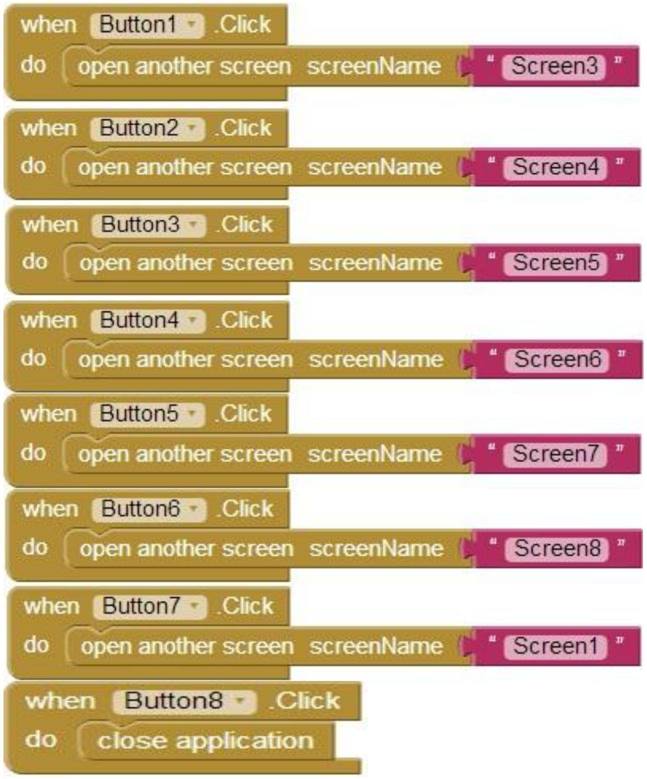


```
when Button1 .Click
do   open another screen  screenName  " Screen2 "

when Button2 .Click
do   close application
```

Figure 11 Typical block editors for Screen 1 (Homepage) of HSS

(b) Screen 2 (Table of Contents):

In this screen Button 1 to 6 are designed for user click for moving to Screen 3 to 8, while Button 7 and 8 for returning to home page and exit app, respectively.

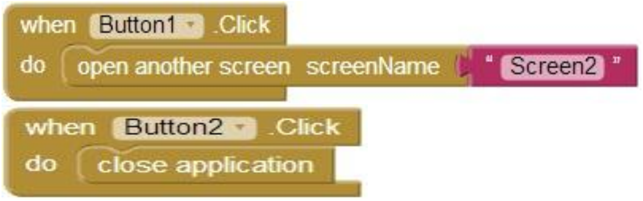


```
when Button1 .Click
do   open another screen screenName "Screen3"

when Button2 .Click
do   open another screen screenName "Screen4"

when Button3 .Click
do   open another screen screenName "Screen5"

when Button4 .Click
do   open another screen screenName "Screen6"

when Button5 .Click
do   open another screen screenName "Screen7"

when Button6 .Click
do   open another screen screenName "Screen8"

when Button7 .Click
do   open another screen screenName "Screen1"

when Button8 .Click
do   close application
```

Figure 12 Block editors for Screen 2 (Table of Contents) of HSS

(c) Screen 3 (Introduction):

In this screen the introductory data are shown by the use of Labels item by item and only two buttons designed for moving to Screen 2 (table of contents) and exit of app.

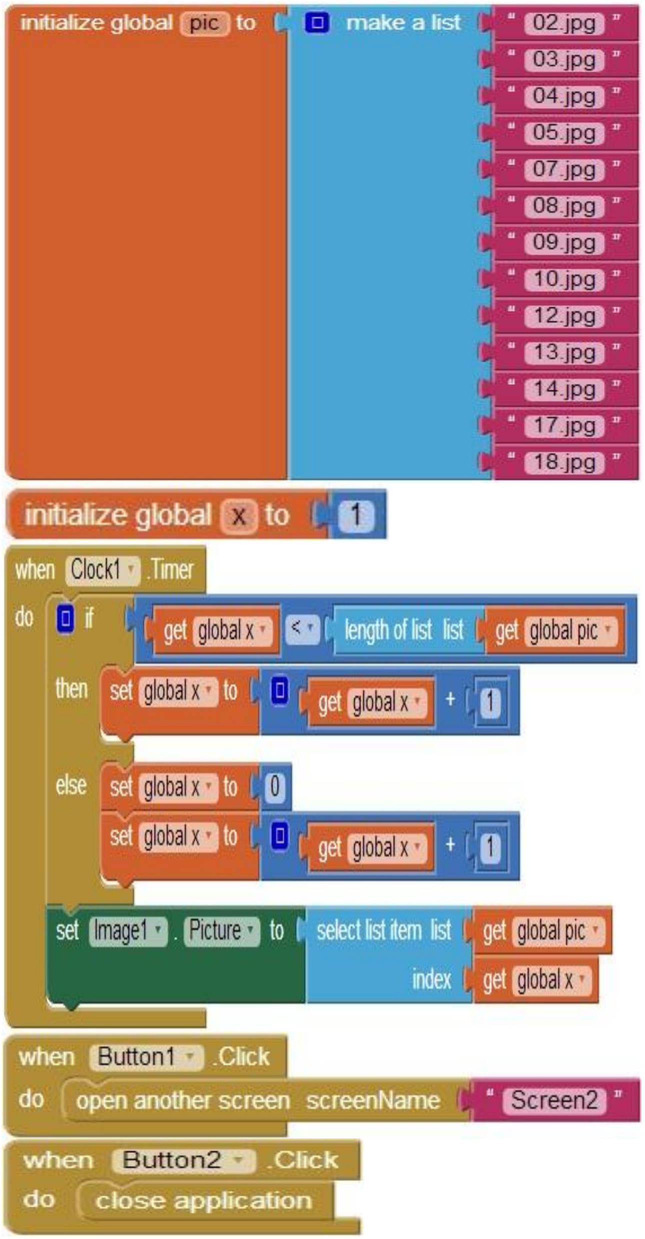


```
when Button1 .Click
do   open another screen screenName "Screen2"

when Button2 .Click
do   close application
```

Figure 13 Block editors for Screen 3 (Introduction) of HSS

(d) Screen 4 (Environment of the House):

Since this screen is designed for picture show of the house, we employ array to store the pictures and use Timer event of Clock to control the changing pictures.

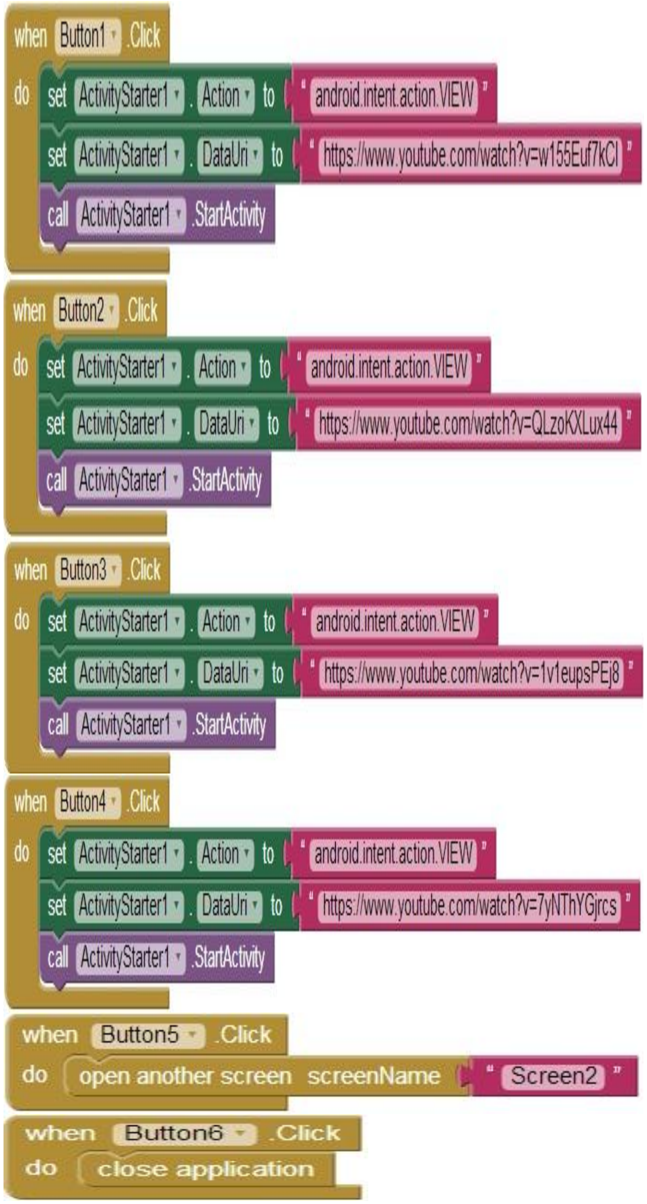


```
initialize global pic to   make a list "02.jpg"
                                       "03.jpg"
                                       "04.jpg"
                                       "05.jpg"
                                       "07.jpg"
                                       "08.jpg"
                                       "09.jpg"
                                       "10.jpg"
                                       "12.jpg"
                                       "13.jpg"
                                       "14.jpg"
                                       "17.jpg"
                                       "18.jpg"

initialize global x to 1

when Clock1 .Timer
do   if    get global x < length of list list get global pic
     then  set global x to get global x + 1
     else  set global x to 0
           set global x to get global x + 1
     set Image1 . Picture to select list item list get global pic
                                              index get global x

when Button1 .Click
do   open another screen screenName "Screen2"

when Button2 .Click
do   close application
```

Figure 14 Block editors for Screen 4 (Environment of the House) of HSS

(e) Screen 5 (Movie Show of the House):

In the Screen 5 for movie show of the house we employ the ActivityStarter to activate the movie play of the movie file stored previously on YouTube. In this way a lot of designed and uploaded multi-media mp4 files, which usually are big files, can be tracked conveniently by smart phones.

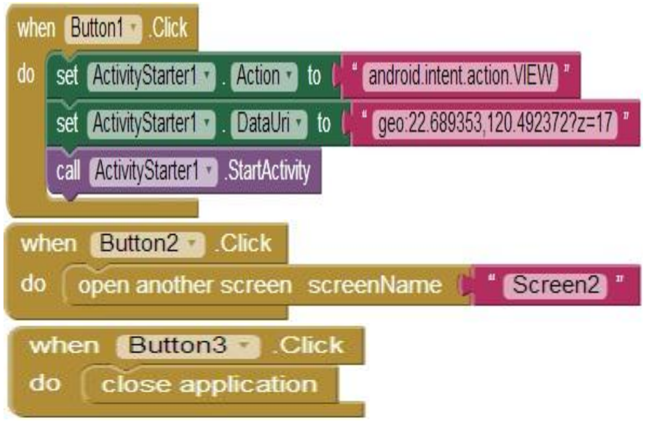


```
when Button1 .Click
do  set ActivityStarter1 . Action to " android.intent.action.VIEW "
    set ActivityStarter1 . DataUri to " https://www.youtube.com/watch?v=w155Euf7kCl "
    call ActivityStarter1 .StartActivity

when Button2 .Click
do  set ActivityStarter1 . Action to " android.intent.action.VIEW "
    set ActivityStarter1 . DataUri to " https://www.youtube.com/watch?v=QLzoKXLux44 "
    call ActivityStarter1 .StartActivity

when Button3 .Click
do  set ActivityStarter1 . Action to " android.intent.action.VIEW "
    set ActivityStarter1 . DataUri to " https://www.youtube.com/watch?v=1v1eupsPEj8 "
    call ActivityStarter1 .StartActivity

when Button4 .Click
do  set ActivityStarter1 . Action to " android.intent.action.VIEW "
    set ActivityStarter1 . DataUri to " https://www.youtube.com/watch?v=7yNThYGjrcs "
    call ActivityStarter1 .StartActivity

when Button5 .Click
do  open another screen  screenName " Screen2 "

when Button6 .Click
do  close application
```

Figure 15 Block editors for Screen 5 (Movie Show of the House) of HSS

(f) Screen 6 (Map of the House):

In this screen we design an Image for showing the outlook of the house and adopt Google map to show the map of house.

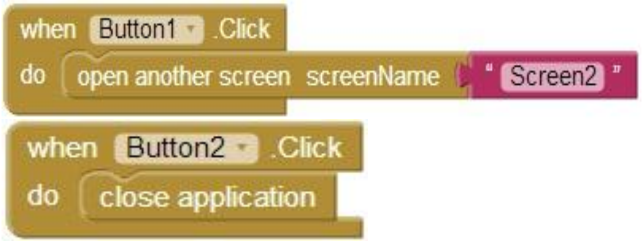


```
when Button1 .Click
do  set ActivityStarter1 . Action to " android.intent.action.VIEW "
    set ActivityStarter1 . DataUri to " geo:22.689353,120.492372?z=17 "
    call ActivityStarter1 .StartActivity

when Button2 .Click
do  open another screen  screenName " Screen2 "

when Button3 .Click
do  close application
```

Figure 16 Block editors for Screen 6 (Map of the House) of HSS

(g) Screen 7 (Top View of Each Floor):

The top view of each floor is very convenient for buyers to understand the layout design of each room. We show this picture by the use of Image.

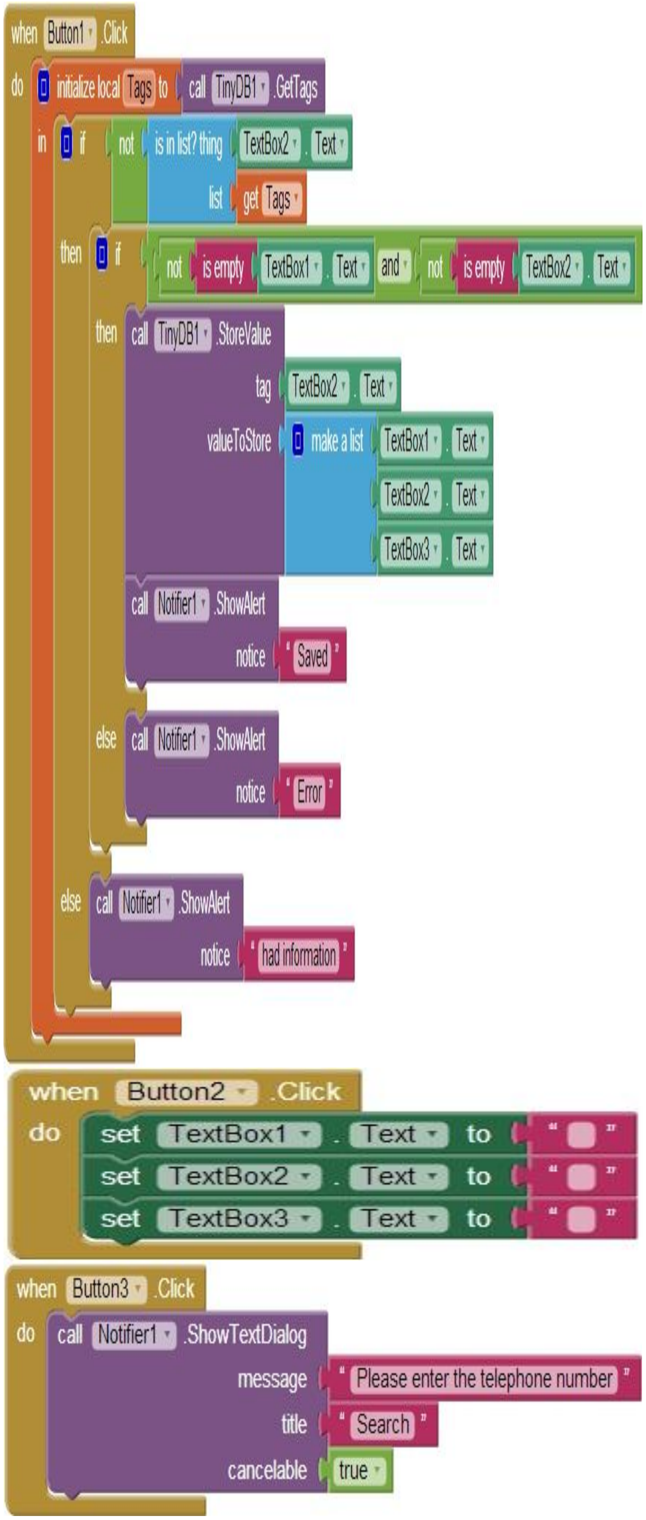


```
when Button1 .Click
do  open another screen  screenName " Screen2 "

when Button2 .Click
do  close application
```

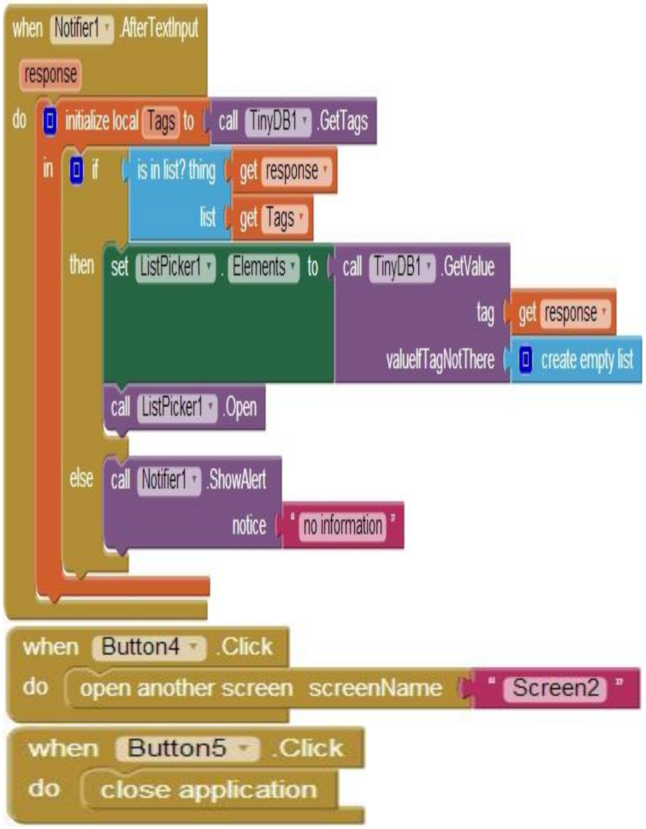
Figure 17 Block editors for Screen 7 (Top View for Each Floor) of HSS

(h) Screen 8 (Data of Potential Buyers):

In this module the design key point is to input and save the data obtained from the interview between seller and buyer. Data of potential buyers can be recorded by another ways independently, but it is convenient for the sellers if the App contains this function because he/she can record immediately after interview. When a seller finds a potential buyer and got his/her agreement to save the data for future connection. In App Inventor we can employ TinyDB or TinyWebDB to store the data input. In this small app we employ only the TinyDB as shown in Fig. 18.



```
when Button1 .Click
do  initialize local Tags to  call TinyDB1 .GetTags
    in  if  not  is in list? thing  TextBox2 . Text
                                list  get Tags
        then  if  not  is empty  TextBox1 . Text  and  not  is empty  TextBox2 . Text
              then  call TinyDB1 .StoreValue
                                   tag  TextBox2 . Text
                          valueToStore  make a list  TextBox1 . Text
                                                     TextBox2 . Text
                                                     TextBox3 . Text
                    call Notifier1 .ShowAlert
                                notice  " Saved "
              else  call Notifier1 .ShowAlert
                                notice  " Error "
        else  call Notifier1 .ShowAlert
                          notice  " had information "

when Button2 .Click
do  set TextBox1 . Text to  " "
    set TextBox2 . Text to  " "
    set TextBox3 . Text to  " "

when Button3 .Click
do  call Notifier1 .ShowTextDialog
                 message  " Please enter the telephone number "
                   title  " Search "
              cancelable  true
```

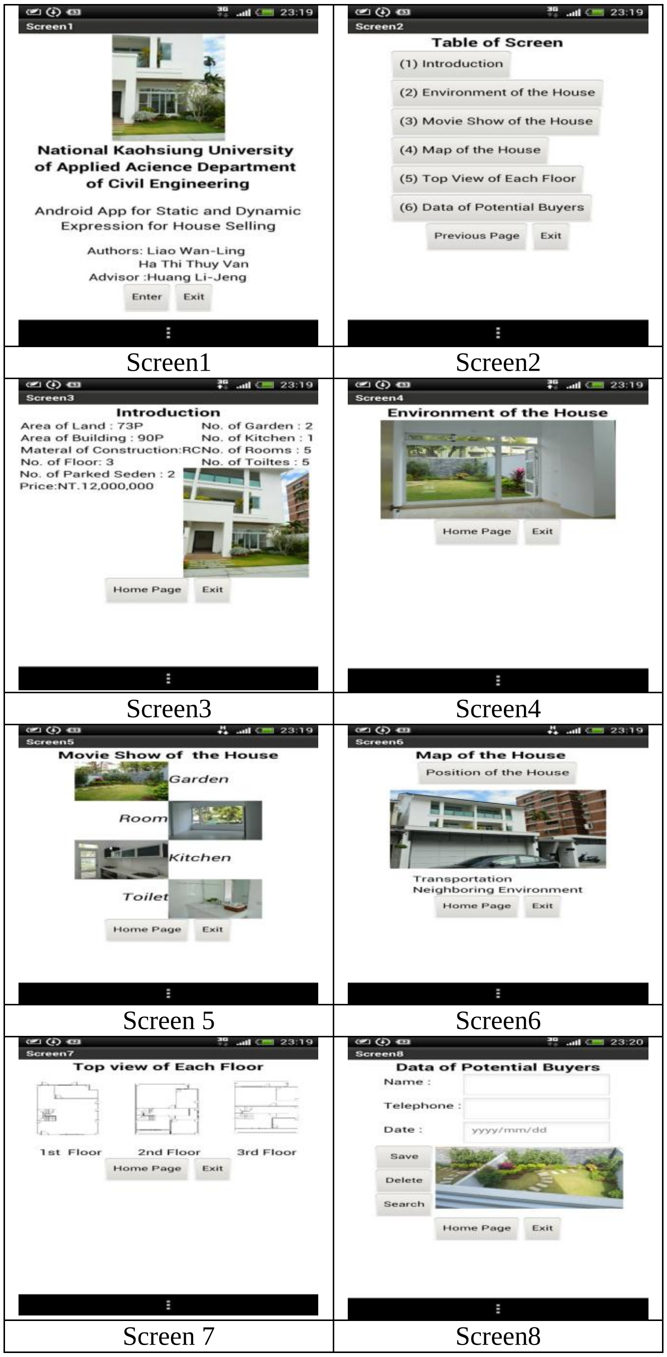


```
when Notifier1 .AfterTextInput
  response
do  initialize local Tags to  call TinyDB1 .GetTags
    in  if  is in list? thing  get response
                         list  get Tags
        then  set ListPicker1 . Elements to  call TinyDB1 .GetValue
                                                            tag  get response
                                             valueIfTagNotThere  create empty list
              call ListPicker1 .Open
        else  call Notifier1 .ShowAlert
                          notice  " no information "

when Button4 .Click
do  open another screen  screenName  " Screen2 "

when Button5 .Click
do  close application
```

Figure 18 Block editors for Screen 8 (Data of Potential Buyers) of HSS

C3. Testing and Deployment Phase

The multi-screens can be tested on the PC via emulator if no smart phone at hands or deployed as HSS.apk and installed on smart phone. In this paper the results obtained from smart phone are shown in Fig. 19 and Fig.20.



Screen1

National Kaohsiung University of Applied Acience Department of Civil Engineering

Android App for Static and Dynamic Expression for House Selling

Authors: Liao Wan-Ling
Ha Thi Thuy Van
Advisor :Huang Li-Jeng

Screen2

Table of Screen

(1) Introduction
(2) Environment of the House
(3) Movie Show of the House
(4) Map of the House
(5) Top View of Each Floor
(6) Data of Potential Buyers

Screen3

Introduction

Area of Land : 73P
Area of Building : 90P
Materal of Construction:RC
No. of Floor: 3
No. of Parked Seden : 2
Price:NT.12,000,000
No. of Garden : 2
No. of Kitchen : 1
No. of Rooms : 5
No. of Toiltes : 5

Screen4

Environment of the House

Screen 5

Movie Show of the House

Screen6

Map of the House

Transportation
Neighboring Environment

Screen 7

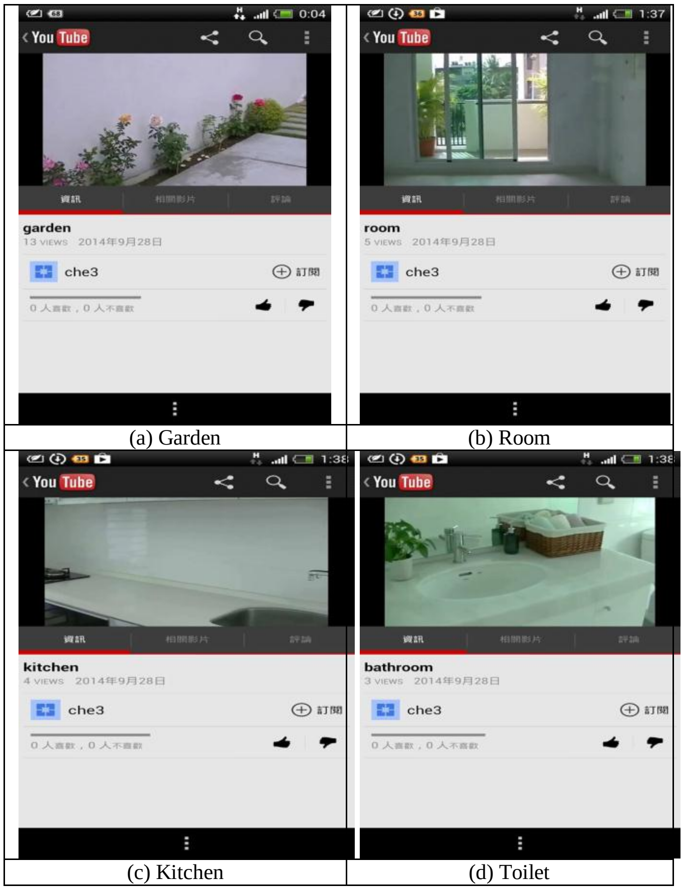
Top view of Each Floor

Screen8

Data of Potential Buyers

Name :
Telephone :
Date : yyyy/mm/dd

Figure 19 Typical results from smart phone of HSS



(a) Garden (b) Room

(c) Kitchen (d) Toilet

Figure 20 Typical results from smart phone of sub-pictures in Screen 4 of HSS (a) garden, (b) room, (c) kitchen, and (d) toilet

We also can design a special meaningful picture for the image of icon for the app HSS as shown in Fig. 21.

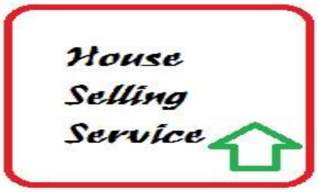


Figure 21 Image picture for app HSS

III. Discussion

A. System Performance

The first version of HSS has been completed and executable on smart phone. The specifications of the smart phone we employed for testing HSS can be observed in TABLE I

TABLE I
SPECIFICATION OF THE SMART PHONE TESTING HSS

Item	Specification
Smart Phone	hTC ONE X+
Operating System	Android OS 4.1 Jelly Bean with hTC Sense 4+
CPU	1.7 GHz
Screen Display	4.7” Super LCD2 720x1280 HD (Gorrila Glass 2)
Storage	1GB RAM, 64 GB ROM
External Storage	Micro SIM
Micro USB	2.0
Audio Support	Mp3, mid, wav, wma, aac, amr, ogg, m4a
Video Support	3gp, 3g2, mp4, wmv, avi
Batery	2100 mAh
Dimension	134.36 × 69.9 × 8.9mm
Weight	135 g

The results are accepted as expected. Although it is only a small one, it provides a hybrid way combined with static and dynamic expression for a new house for selling. It is easy to learn, to use and to teach. The most important feedback from the users is its easy portability as compared with PC or Notebooks, especially house sellers usually need move from one floor to another floor, from one house to another.

B. Advantages

The comparison of different expressions between using App of HSS on smart phones and three-dimensional house models can be referred to Table II.

TABLE II
COMPARISON OF HSS APP AND 3D HOUSE MODELS

Features	HSS App Run on Smart Phones	3D House Models
Feature	Virtual	Real
Cost	Low	High
Preparation Time	Short	Long
Portability	High	Low
Expression	Static/Dynamic	Static
Re-newness	Easy	Difficult
Machine Dependence	Yes	No
Maintanance	Ease	Difficult

IV. CONCLUDING REMARKS

Application of MIT App Inventor to design Android App for static and dynamic expression for visual aids in house selling service executable on Android platforms such as smart phones and handheld pads has been conducted successfully. System development procedures, layout design and program coding will be discussed. A typical new built selling project in Ping-Tung of Taiwan was employed to show the results of design app.

The success of this attempt shows that App Inventor can help engineers and designers develop their own specific apps run on conveniently portable platforms. The development procedure for HSS in this paper can be extended for many projects of house selling of different buildings, such as commercial, business and apartments.

REFERENCES

[1] Windermere Services Company. 2008. Windermere Real Estate: Guide to Selling a House

[2] Mann, D., O Cathain,, C. 2001. 40 Inventive (Architecture) Principles with Examples. The TRIZ J.

[3] Gennady, R. 2005. 40 Inventive Principles in Marketing, Sales and Advertising.

[4] Dwyer, F. R., Schurr, P. H. and Oh, S. 1987. Developing Buyer-Seller Relationships. J. Marketing, 51, 11-27.

[5] Atul, P. and Jagdish, N. S. 2001. Customer Relationship Management: Emerging Practice, Process, and Discipline. J. Eco .and Soc. Res., 3(2), 1-34.

[6] Jean-Michel, M. and Ganael B. 2009. Efects of Sales Force Automation Use on Sale Force Activities and Customer Relationship Management Process. J. Person. Sell. & Sales Mana., 28(2), 167-184.

[7] Consumer Affairs Victoria. 2008. Real estate: A Guide for Buyers and Sellers.

[8] Tyler, J. 2011. App Inventor for Android. Baker & Tayl, 2011.

[9] Wolber, D., Abelson, H., Spertus, E., andLooney, L. 2011. App Inventor. O’Reilly Media.

[10] Kloss-Jorg, H. 2013. Android Apps with App Inventor:The Fast and Easy Way to Build Android Apps. Addison-Wesley, 2013.

[11] Lakshmi, P., Jeffery H., and Alex W. 2014. Android App Inventor for the Absolute Beginner. Cengage Learning PTR, Boston, USA.

[12] Huang, L. J. 2014. Object-Oriented Programming for Android Using App Inventor. Note for Lecture, National Kaohsiung University of Applied Science, Taiwan, R. O. C. (in Chinese)

[13] Wikipedia. 2014. App Inventor for Android. http://en.wikipedia.org/wiki/Google_App_Inventor.

[14] Wolber, D. 2011. App Inventor and Real-World Motivation. Proc. 42th ACM Tech. Symp. Comp. Sci. Edu. (SIGCSE ’11), ACM, NY, 601-606.

[15] Hsu, Y. C., Rice, K., and Dawley, L. 2012. Empowering Educators with Google’s Android App Inventor: An Online Workshop in Mobile App Design. British J. Educa. Tech. 43(1), E1-E5.

[16] Arachchilage, N. A. G., Love, S.and Scott, M. 2012. Designing a Mobile Game to Teach Conceptual Knowledge of Avoiding “Phishing Attacks”, Int. J. e-Learning Secur., 2(2), 127-132.

[17] William, L. H. 2013. Teaching and Assessing Programming Fundamentals for Non Majors with Visual Programming. Proc. 18th ACM Conf. Inno. Tech. Compu. Sci. Edu. (ITiCSE ’13), ACM, NY, USA.

[18] Zhang, C. 2014. Healthcare Mobile App Development with App Inventor in a Health IT Course. Proc. South. Asso. Inform. Sys. Conf., Macon, GA, USA, March 21-22, 2014.

[19] HuiRu Shih, P. E., Jackson, J. M., Wilson, C. L. H. and Yuan, P. C. 2014. Using MIT App Inventor in an Emergency Management Course to Promote Computational Thinking. 121st ASEE Annual Conference & Exposition, June 15-18, Paper ID: #8719.

[20] Huang, L. J., Chiu, Y. C., Hung, Y. T. and Jian, Y. S. 2014. Android Programming Design for Matrix Method of Static Structural Analysis on Smart Phones Using App Inventor 2. International Journal of Recent Development in Engineering and Technology (IJRDET), 3(3), 85-96.

Appendix: Conceptual design of the App of Home Selling Service (HSS)

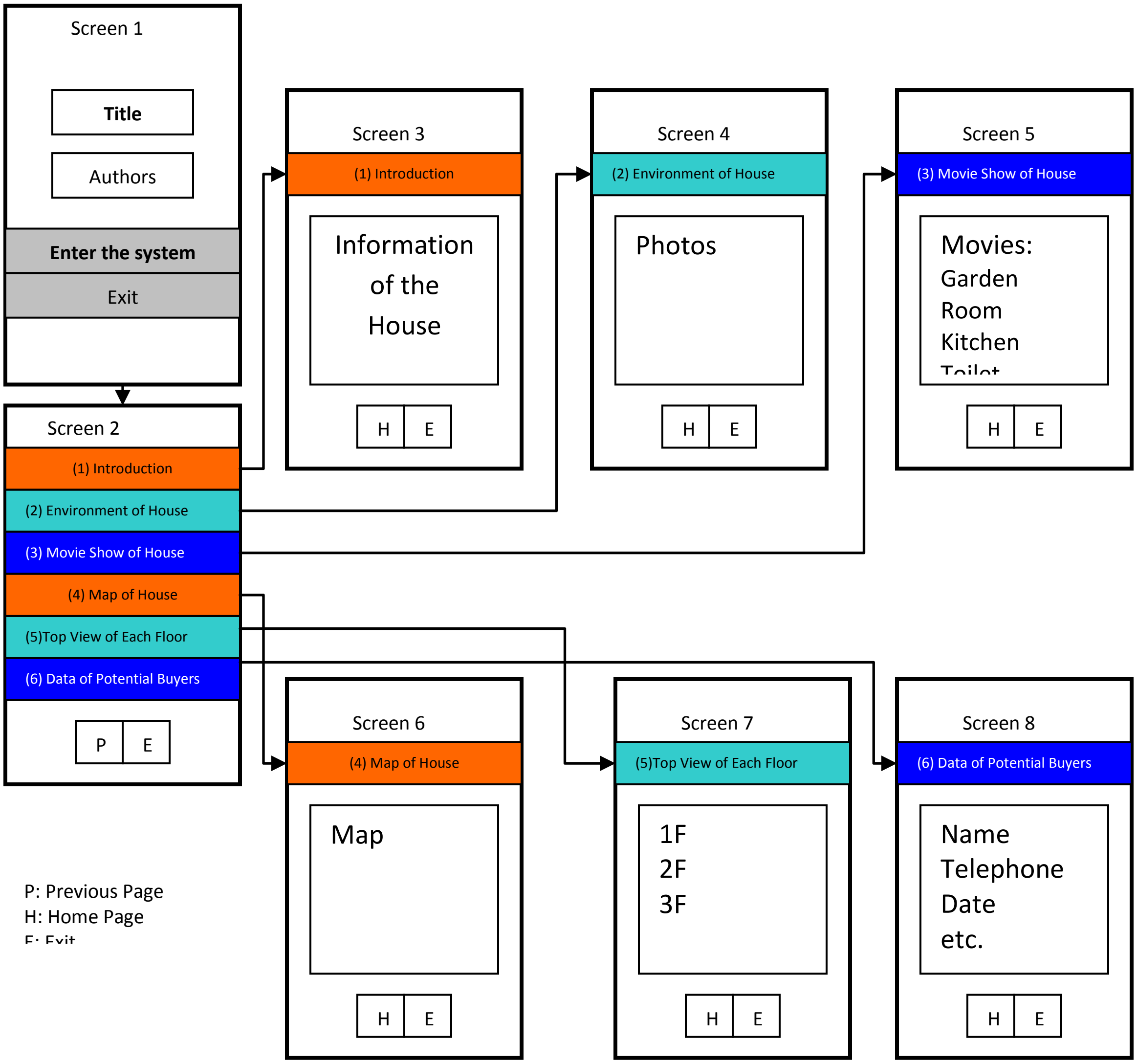


Figure A Conceptual design of the App of Home Selling Service (HSS)