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

Android Programming Design for Matrix Method of Static Structural Analysis on Smart Phones Using App Inventor 2

Huang Li-Jeng[1], Chiu Yu-Chuan[2], Hung Yu-Ting[2], Jian Ya-Shin[2]

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

***Abstract*—App Inventor and its new version, App Inventor2, are no-code open platform for Android mobile app development using drag-and-drop approach and visual programming environment. They provide easy environments for people to create apps executable on Android platforms such as smart phones and handheld pads. This paper is aimed at Android programming for professional application, i.e., matrix method of static structural analysis (MMSSA). System development procedures, layout design and program coding will be explained. A typical numerical example of static analysis of statically in-determinated continuous beam will be employed to show the results of analysis and plotting such as the shearing force diagram, bending moment diagram, rotation angle diagram and displacement diagrams. These plots will be compared with those obtained by the counterpart developed using EXCEL with VBScript and run on the desktop PC.**

***Keywords*—Android Apps, App Inventor2, Matrix Methods, Smart Phones, Static Structural Analysis.**

I. INTRODUCTION

Structural analysis is an important skill and necessary technology for students and practician in civil engineering. Classical structural analysis has become a necessary profession for civil engineers [1-3]. Nowadays matrix methods of structural analysis or finite element method (FEM) has become a major trend in developing the numerical and computational tools for structural analysis [4-8]. In the static analysis of a structure the fundamental purpose is to calculate the internal forces (e.g. axial force, shearing force and bending moment for a beam) and kinematic quantities (external displacements, rotations and internal deformations, etc.) [1-3]. There are many famous commercial professional FEM-based softwares to help people to conduct structural analysis, static and dynamic, such as SAP2000, ETABS, ANSYS, NASTRAN, etc. However, advanced structural engineers usually only require to treat a small project and they attempt to code their programs for analysis. Traditionally they can code in specific programming, such as BASIC, FORTRAN, MATLAB, MATHEMATICA, etc. and execute on PC.

The disadvantage is loss of time efficiency and task convenience. It is very interesting to structural engineers if there are structural analysis tools on smart phones or handheld pads.

Recently there are many approaches to develop and design Android apps: (1) Java JDK with NetBeans (or Eclipse); (2) HTML, CSS with JavaScript; (3) Corona SDK (also applicable for iOS platform); (4) PhoneGap; (5) App Accelerator; (6) Unity 3D; (7) App Inventor (and its new version App Inventor 2), (8) Scratch, (9) Titanium, (10) AppMobi, (11) Basic4Android, etc. Each tool possesses its inherent special features and advantages/disadvantages and Android programmers have their independent and personal appreciation.

App Inventor and its new version, App Inventor2, are tools to design Apps for the Android platform. They are developed originally by Google and then maintained by MIT. 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. QR-code, 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 [9-13].

The major elements in App Inventor 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.

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

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) [14], 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.[15-20].

This paper presents a trial of application of App Inventor2 to design Android App for static structural analysis executable on Android platforms such as smart phones and handheld pads. System development procedures, layout design and program coding will be discussed. A typical numerical example of static analysis of continuous beam will be employed to show the results of analysis including the internal forces (shearing force diagram and bending moment diagram) as well as external deformations (rotation angle diagram and displacement diagram). These plots will be compared with those obtained by the counterpart developed using EXCEL with VBScript.

II. Development Of The System

A. Design Procedure of Android App for Static Structural Analysis

The design procedure of development of Android App for static structural analysis can be observed as shown in Fig.1.

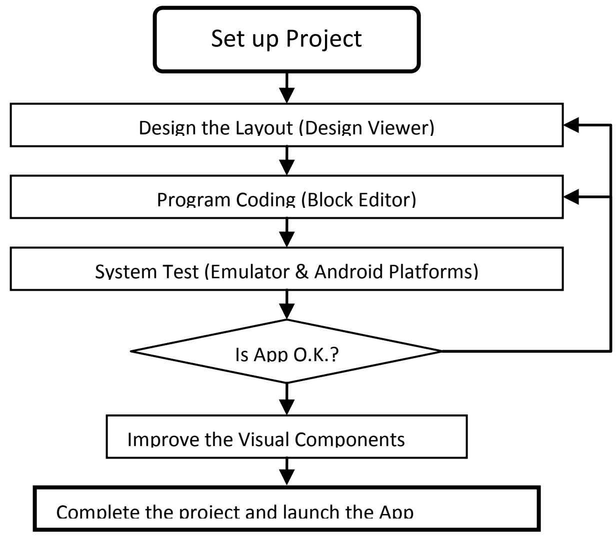


Figure 1 Flow chart of Android App system development using App Inventor 2

B. Design Procedure of for Static Structural Analysis

We choose direct element (stiffness) method, which can also be called the coding method to design the mobile application. The procedure can be summarized in the following procedure [8]:

(1) Define nodal and element data;

(2) Initiate global displacement and force vectors $\{D\},\{F\}$ (2 D.O.F. for each node);

(3) Initiate element displacement and force vectors $\{d^{(e)}\},\{f^{(e)}\}$ (4 D.O.F. for each element); the sequence is $\{d_1, r_1, d_2, r_2\}$ and $\{V_1, M_1, V_2, M_2\}$, respectively. Displacement and shear are positive upwards while rotation and moment are positive counter-clockwise.

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

(4) Form element stiffness matrix for each element:

$$[k^e] = (\frac{EI}{L})^e \begin{bmatrix} 12/L^2 & 6/L & -12/L^2 & 6/L \\ & 4 & -6/L & 2 \\ & & 12/L^2 & -6/L \\ Symm. & & & 4 \end{bmatrix} \quad (1)$$

where E denotes the Young's modulus, I the moment of inertia, and L the length of the member;

(5) Form the equivalent nodal force vector $\{f_E{}^{(e)}\}$ and added on to direct nodal force vector $\{f^{(e)}\}$;

(6) Assemble element stiffness matrices and element force vectors to global stiffness matrix and global force vector, respectively,

$$[K] = \sum_{e=1}^{NE} [k^e] = [k^1] + [k^2] + \cdots + [k^{NE}] \quad (2a)$$

$$\{F\} = \sum_{e=1}^{NE} \{f^e\} = \{f^1\} + \{f^2\} + \cdots + \{f^{NE}\} \quad (2b)$$

(7) Obtain the system equations of equilibrium:

$$[K]\{D\} = \{F\} \quad (3)$$

(8) Solve the unknown displacement vectors and reactions vector after substituting prescribed end conditions by the way of matrix partitioning technique:

$$\begin{bmatrix} [K_{uu}] & [K_{us}] \\ [K_{su}] & [K_{ss}] \end{bmatrix} \begin{Bmatrix} \{D_u\} \\ \{D_s\} \end{Bmatrix} = \begin{Bmatrix} \{F_u\} \\ \{F_s\} \end{Bmatrix} \quad (4)$$

where the subscript u and s denotes unknown and specified, respectively. From (4) we can obtain

$$\{D_u\} = [K_{uu}]^{-1}(\{F_u\} - [K_{us}]\{F_s\}) \quad (5a)$$

$$\{F_s\} = [K_{su}]\{D_u\} + [K_{ss}]\{D_s\} \quad (5b)$$

(9) Get element displacement vector and element force vector by relating element displacement vectors $\{d^{(e)}\}$ to global displacement vector $\{D\}$ and

$$\{f^{(e)}\} = [k^{(e)}]\{d^{(e)}\} - \{f_E{}^{(e)}\} \quad (6)$$

(10) Calculate and draw shear force (*V*), bending moment (*M*), rotation angle (*R*), displacement (*D*) diagrams.

The procedure for a general static structural analysis can be referred in Fig. 2.

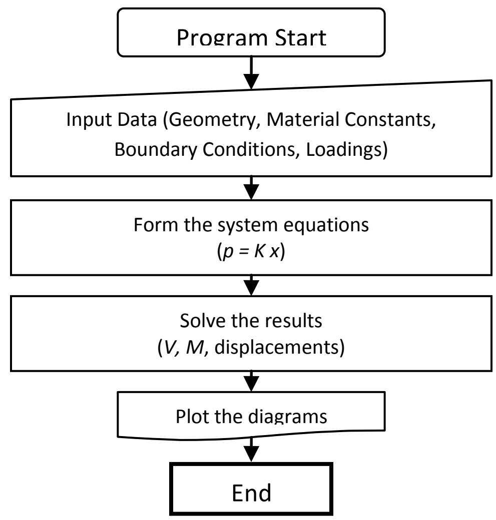


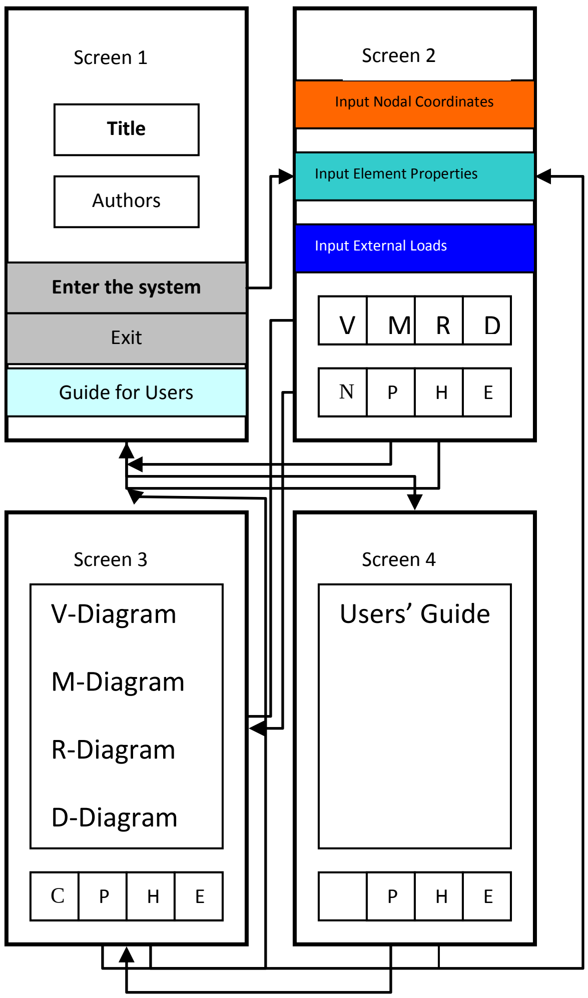
Figure 2 Flow chart of matrix method of static structural analysis of continuous beam system

C. App Design

C1. Conceptual Design Phase

We design the app of matrix method of static structural analysis to be one with multi-screens framework and assign it a name MMSSA. The conceptual design of the system can be shown in Fig. 3. Some detailed arrangements might be changed and modified during the Preliminary Design Phase (Layout Design Phase) and rhe Detailed Design Phase (Programming Coding Phase).

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



V: Shear
M: Moment
R: Rotation
D: Displacement

N: Next Page
P: Previous Page
H: Home Page
E: Exit
C: Clear

Figure 3 Conceptual design of the App of Matrix Method for Static Structural Analysis (MMSSA)

C2. Layout Design Phase

We employ the multi-screen concept to design this system with 4 layouts via *design viewer of App Inventor2*. These 4 layout design can be observed from Fig. 4 to Fig. 7. In each viewer screen the object types employed are:

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

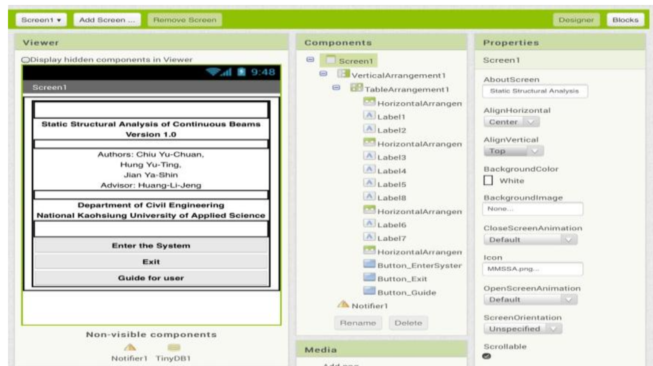


Figure 4 Screen 1(title page) of the Android app MMSSA

(b) Screen 2 (Input Page):

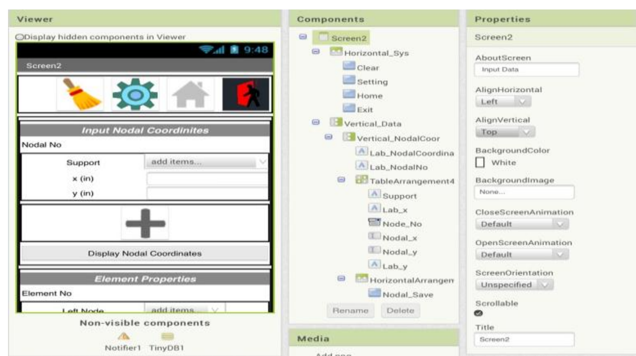


Figure 5 Screen 2 (Input Page) of the Android app MMSSA

(c) Screen 3 (Output Page):

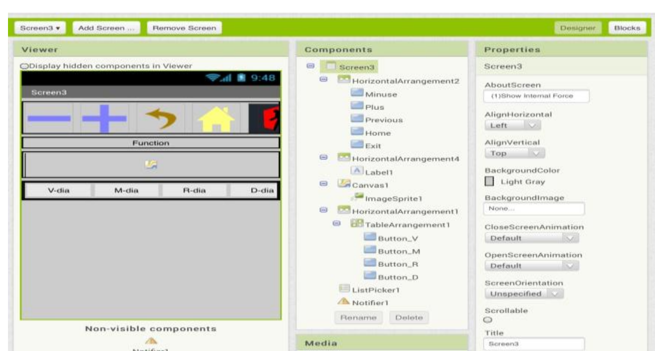


Figure 6 Screen 3 (Output Page) of the Android app MMSSA

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

(d) Screen 4 (User's Guide):

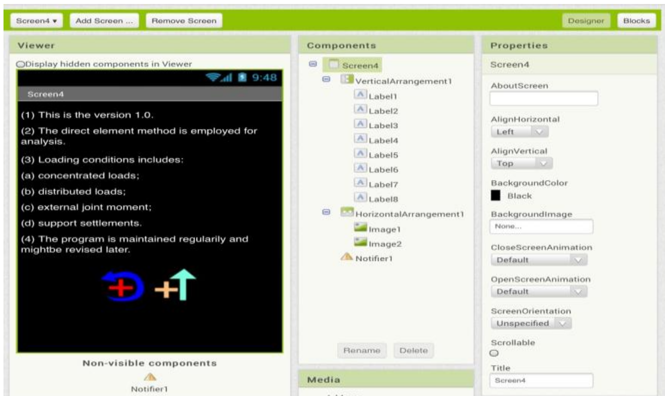


Figure 7 Screen 4 (Users' Guide) of the Android app MMSSA

C3. Program Coding Phase

One of special feature of App Inventor2 is the programming mode the same as building blocks as shown from Fig. 8 to Fig. 11. Some core programs for matrix addition, multiplication and inversion can be referred to Appendix.

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

Figure 8 Typical block editors for Screen 1 (Homepage) of MMSSA

(b) Screen 2 (Input Page):

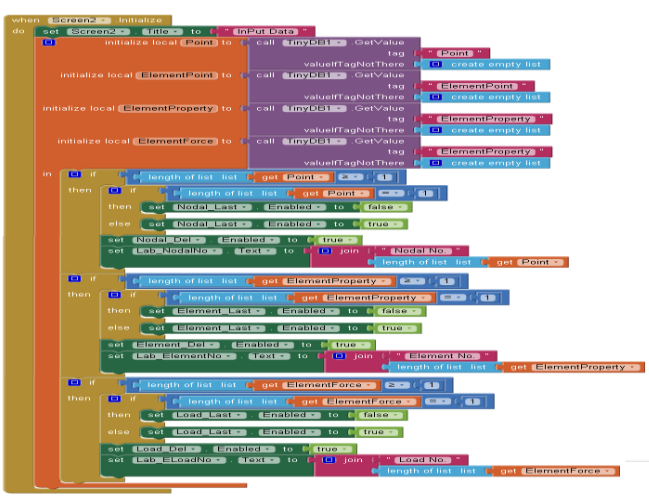


Figure 9 The beginning part of block editors for Screen 2 (Input Page) of MMSSA

(c) Screen 3 (Output Page):

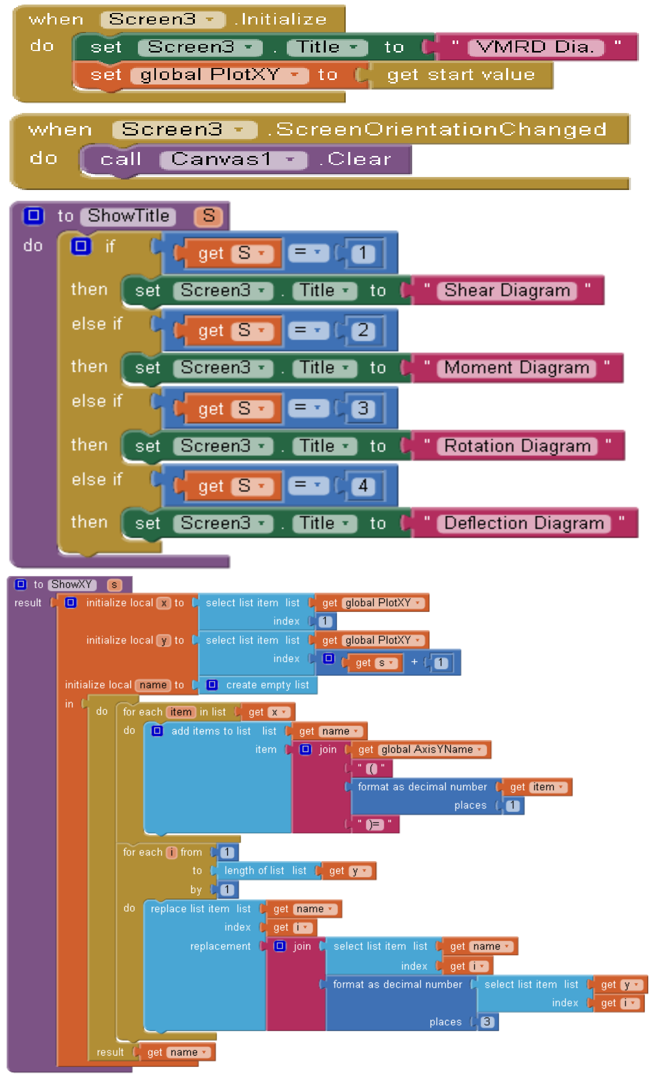


Figure 10 Parts of block editors for Screen 3 (Output) of MMSSA

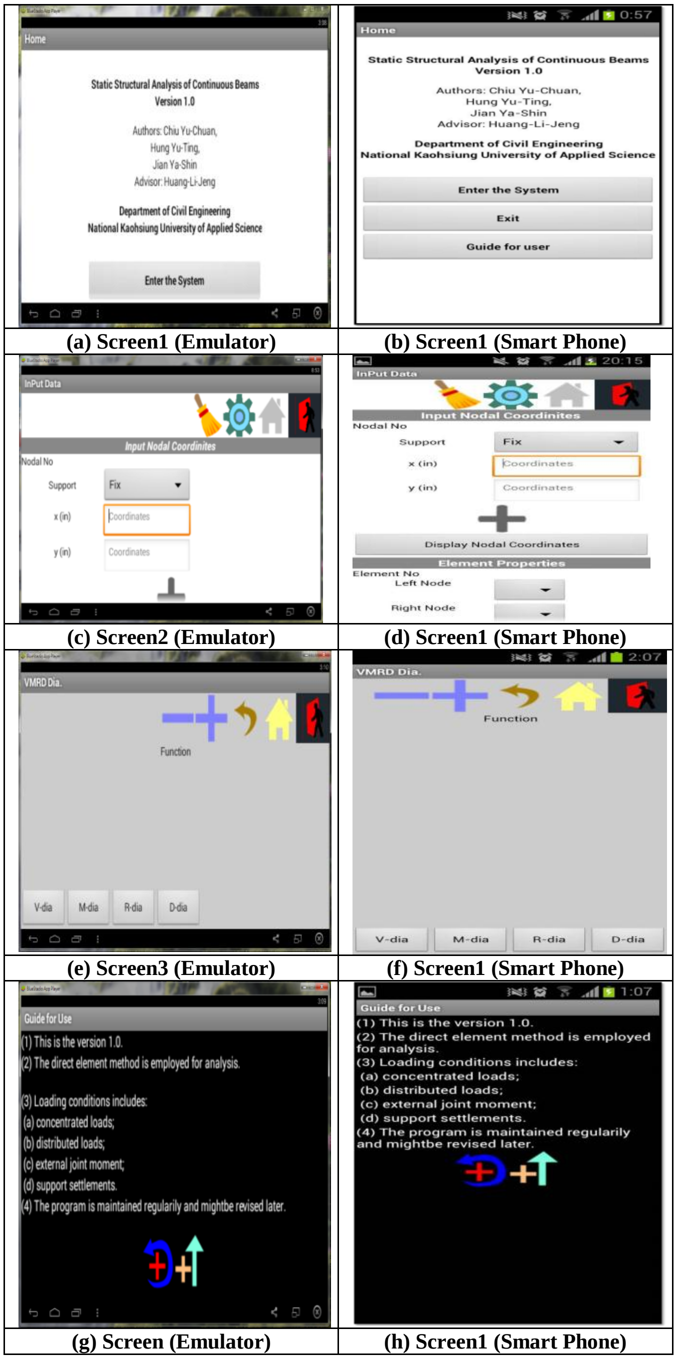
(d) Screen 4 (User's Guide):

Figure 11 Block editors for Screen 4 (Users' Guide) of MMSSA

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 MMSSA.apk and installed on smart phone. The results obtained from emulator and smart phone are shown in Fig. 12.

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



(a) Screen1 (Emulator) (b) Screen1 (Smart Phone)
(c) Screen2 (Emulator) (d) Screen1 (Smart Phone)
(e) Screen3 (Emulator) (f) Screen1 (Smart Phone)
(g) Screen (Emulator) (h) Screen1 (Smart Phone)

Figure 12 Typical results from Emulator and smart phone of MMSSA

We also can design a special meaningful picture for the image of icon for the app MMSSA as shown in Fig. 13.

Figure 13 Image picture for app MMSSA

III. Case Study And Results

A. Problem Description

A typical static structural analysis case was selected for validation test of the App development. The strucrural system is a statically in-determinated continuous beam subjected to various external loadings. The schematic diagram, data of beam length, sectional properties loadings and support conditions are shown in Fig. 14.

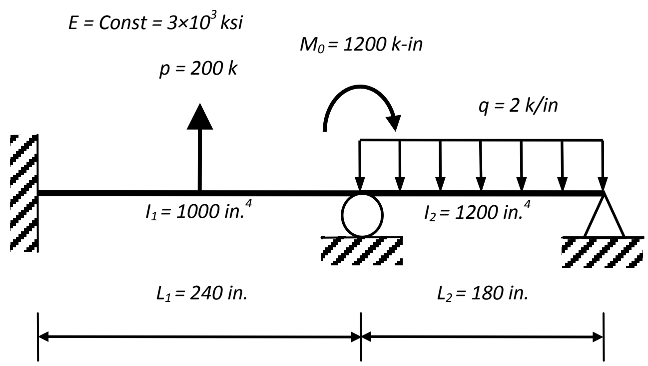


Figure 14 Schematic of static structural analysis of a continuous beam.

B. Input Modules for Android App of Static Structural Analysis

In the execution of designed App some typical input modules are shown in TABLE I. It can be observed that the first homepage is the information of title and authors wherein the account and password mechanism can also be added-on if necessary. The second screen is the interface for basic input data (dimensions of beams, material constants, boundary conditions, external loadings, etc.) It is noticed that combined loading cases are automatically considered. In the programming design fault-input warning mechanism has been included by the use of Notifiers components (which are invisible in AI2) to help users of the App complete the analysis task without occurring the input errors.

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

TABLE I
INPUT DATA OF TEST CASE USING MMSSA

Item		Items	Data
Input Nodal Coordinates	No. 1	x	0
		y	0
	No. 2	x	240
		y	0
	No. 3	x	0
		y	420
Input Element Properties	Element No. 1	Left No.	1
		Right No.	2
		I	1000
		E	3000000
	Element No. 2	Left No.	2
		Right No.	3
		I	1200
		E	3000000
Input Element Loads	Element No. 1	Concentration	200000
	Element No. 2	Distributed	-2000
Input Nodal Loads & Settlement	Nodal No. 1	Point Load	0
		Moment	0
		Settlement	0
	Nodal No. 2	Point Load	0
		Moment	-1200000
		Settlement	0
	Nodal No. 3	Point Load	0
		Moment	0
		Settlement	0

C. Output Modules for Android App of Static Structural Analysis

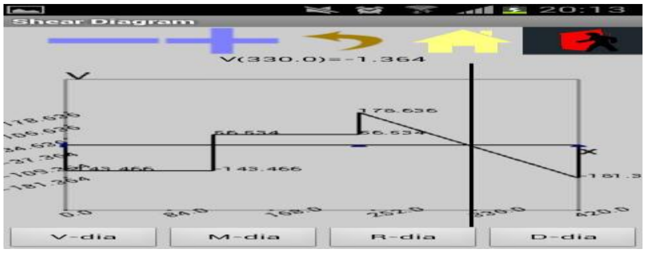
The environment of smart phone we employed for testing MMSSA can be observed in TABLE II.

TABLE II
INPUT DATA OF TEST CASE USING MMSSA

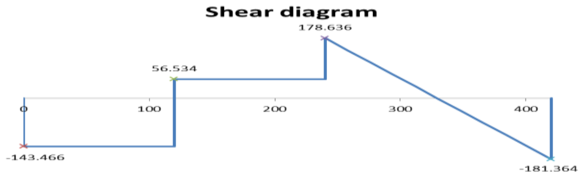
Item	
Smart Phone	GT-i9103
Operating System	Android OS V 2.3
CPU	NVIDIA Tegra 2 1GHz
Screen Display	4.3" LCD 480x800 WVGA
Storage	1GB RAM,16GB ROM
External Storage	Micro SD
USB	2.0
Batery	1650mAh
Dimension	125.3×66.1×8.49mm
Weight	116g

In the execution of designed App some typical output results are shown in Fig. 15 to Fig.18, where the results run by alternate program developed using EXCEL with VBScript by us are also shown for comparison. It can be observed that the results calculated from Android App are rhe same as those obtained from EXCEL macros. Furthermore, in the MMSSA.apk we employed the event ImageSprite. Dragged in App Inventor 2 to let the user can drag the vertical bar to examine values of *V*, *M*, *R*, *D* at each position on beam (the data will also be shown on the top of each screen and changed as the vertical bar moved). The difference is that the structural analysis computed by Android smart phones is now takes more time than that run on PCs or Notebooks installed with Windows OS. This is mainly due to the different infrastructure of CPU design for Android and Windows OSs where floating-point calculation has been improved.

(a) Shear Force Diagram (V-x):



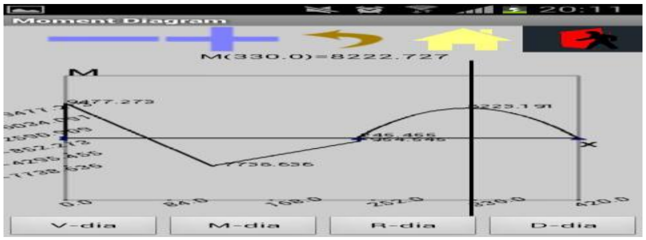
(a)



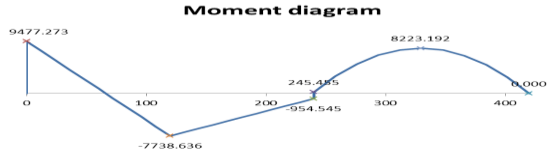
(b)

Figure 15 Output Module (Shear Force Diagram) of the Android App static structural analysis (a) from Android app (b) from EXCEL

(b) Bending Moment Diagram (M-x):



(a)

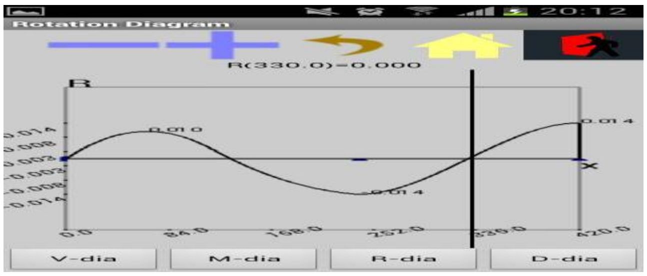


(b)

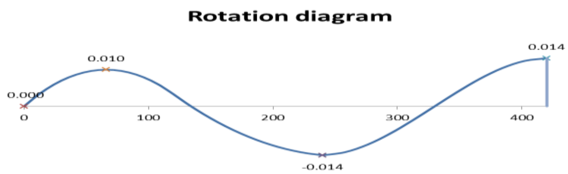
Figure 16 Output Module (Bending Moment Diagram) of the Android App static structural analysis (a) from Android app (b) from EXCEL

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

(c) Rotation Diagram (R-x):



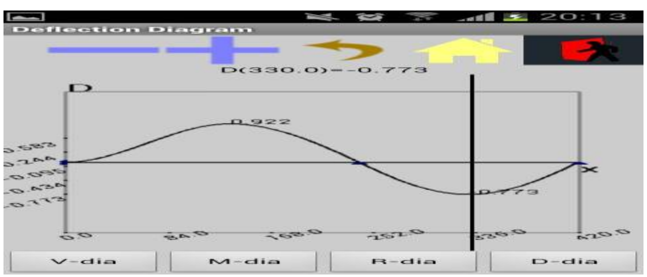
(a)



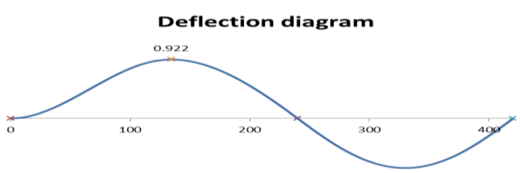
(b)

Figure 17 Output Module (Rotation Diagram) of the Android App static structural analysis (a) from Android app (b) from EXCEL

(d) Displacement Diagram (D-x):



(a)



(b)

Figure 18 Output Module (Displacement Diagram) of the Android App static structural analysis (a) from Android app (b) from EXCEL

It should noticed that the rotation angle diagram and displacement diagram had been enlarged vertically for better reading because their real values are very small in practice.

D. Comparison of Personally Developed Static Structural Analysis Using Android App and EXCEL

The comparison of programs developed personally for static structural analysis of a continuous beam using Android and EXCEL can be referred to Table III.

TABLE III
COMPARISON OF STATIC STRUCTURAL ANALYSIS USING ANDROID AND EXCEL

Features	Android App Run on Smart Phones	EXCEL Run on PC
OS	Android-based	Windows-based
Cost	Free Software	Charged
Execution Speed	Slow	Rapid
Portability	High	Low
Interface	OOP	Non-OOP
Programming Extension	Ease	Ease

IV. CONCLUDING REMARKS

Application of App Inventor2 to design Android App for static structural analysis 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 numerical example of static analysis of continuous beam will be employed to show the results of analysis including the shearing force diagram, bending moment diagram and displacement diagrams. These plots had been compared with those obtained by the counterpart developed using EXCEL with VBScript and agreed well with each other. The success of this attempt shows that App Inventor2 can help engineers develop their own specific apps run on conveniently portable platforms. The development procedure for MMSSA in this paper can be extended for many static structural analyses of different structures, such as trusses, rigid frames and hybrid systems.

REFERENCES

[1] Hsieh, Y. Y. and Mau, S. T. 1998. Elementary Theory of Structures. Prentice-Hall, Englewood Cliffs, New Jersey.

[2] Hsu, D. S. 1996. Theory of Structural Analysis (1) (2). (In Chinese)

[3] Hibbeler, R. C. 1997. Structural Analysis, Prentice-Hall.

[4] Livesley, R. K. 1964. Matrix Methods of Structural Analysis. Pergamon Press.

[5] Bray, K. H. M., Croxton, P. C. L. and Martin, L. H. 1976. Matrix Analysis of Structures. Hodder Arnold, London, UK..

[6] Sack, R. L. 1989. Matrix Structural Analysis PWS-Kent Publishing Company.

[7] Sennett, R. E. 1994. Matrix Analysis of Structures. Prentice-Hall Inc.

[8] Yang, T. Y. 1986. Finite Element Structural Analysis. Prentice-Hall, Inc.

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

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

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

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

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

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

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

[16] 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.

[17] 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.

[18] 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.

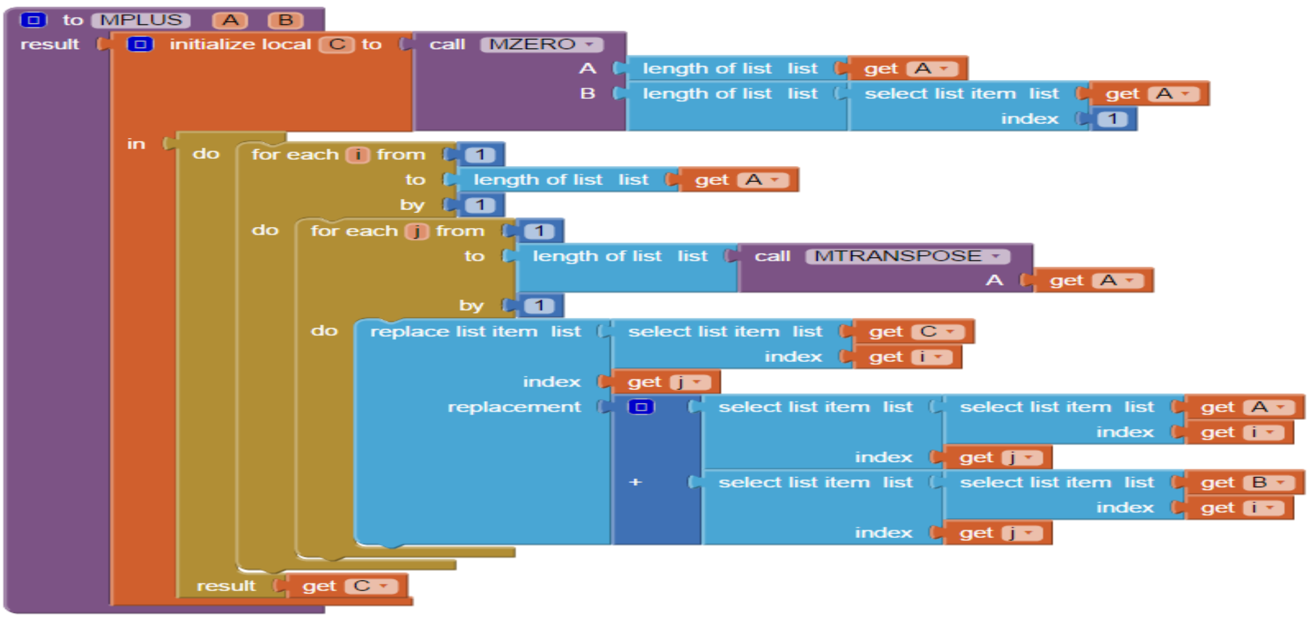
[19] Zhang, C. 2014. Healthcare Mobile App Development with App Inventor in a Health IT Course. Proc. South. Asso. Inform. Sys. Conf., Macon, GA, USA, March 21-22, 2014.

[20] 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.

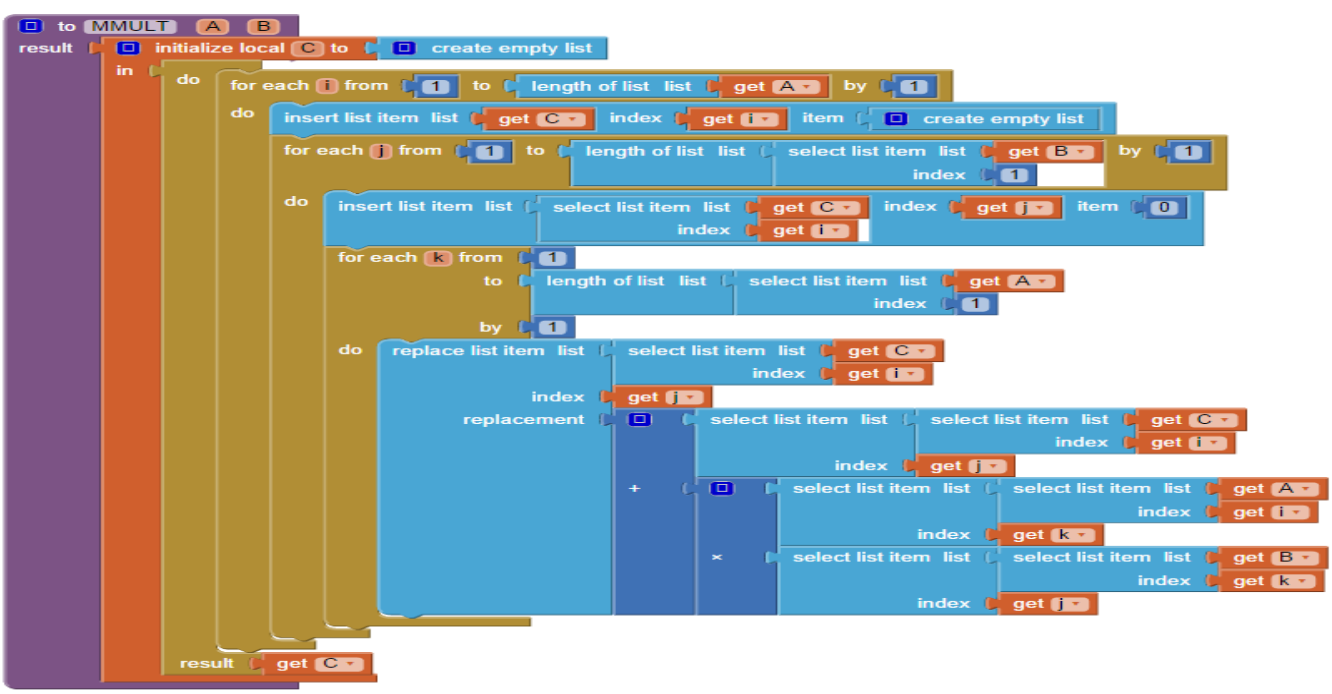
International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 3, September 2014)

Appendix: App Inventor2 Block Editors for Some Matrix Operations employed in Structural Analysis

A.1 Addition of Two matrices:



A.2 Multiplication of Two matrices:



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

A.3 Inversion of stiffness matrix: