Sinographs: Studying Chinese characters through Adobe flash on Apple iPhone

Paul Guo
SINOGRAPHS
Studying Chinese Characters through Adobe Flash on Apple iPhone

Paul Guo

March 7, 2013

Rochester Institute of Technology
Thesis submitted to the Faculty of the Computer Graphics Design, School of Design | College of Imaging Arts and Sciences in candidacy for the degree of Master of Fine Arts
Approvals

Chief Advisor: Shaun Foster, Assistant Professor, Computer Graphics Design

__________________________  ___________________________
Signature of Chief Advisor  Date

Associate Advisor: Chris Jackson, Assistant Professor, Computer Graphics Design

__________________________  ___________________________
Signature of Associate Advisor  Date

Associate Advisor: Marla Schweppe, Professor, Computer Graphics Design

__________________________  ___________________________
Signature of Associate Advisor  Date
Reproduction Granted:

I, __Paul Guo__, hereby grant permission to Rochester Institute of Technology to reproduce my thesis documentation in whole or part. Any reproduction will not be for commercial use or profit.

______________________________
Signature of Author

______________________________
Date

Inclusion in the RIT Digital Media Library Electronic Thesis and Dissertation (ETD) Archive:

I, __Paul Guo__, additionally grant to Rochester Institute of Technology Digital Media Library the non-exclusive license to archive and provide electronic access to my thesis in whole or in part in all forms of media in perpetuity. I understand that my work, in addition to its bibliographic record and abstract, will be available to the worldwide community of scholars and researchers through the RIT DML. I retain all other ownership rights to the copyright of the thesis. I also retain the right to use in future works (such as articles and books) all or part of this thesis. I am aware that Rochester Institute of Technology does not require registration of copyright for ETDs. I hereby certify that, if appropriate, I have obtained and attached written permission statements from owners of each third party copyrighted matter to be included in my thesis. I certify that the version I submit is the same as that approved by my committee.

______________________________
Signature of Author

______________________________
Date
Abstract

Demo URL: http://www.paulguo.com/sinographs

Sinographs are Chinese characters used in the Chinese and Japanese languages. People who are learning Chinese find sinographs challenging to memorize. My objective is to teach 50 of the most-used simplified sinographs to Chinese learners and to help them determine how to form basic characters through an interactive application developed in Flash Professional for utilization on the iPhone and as on a demo webpage. Interactive applications appear to becoming increasingly more popular, especially in web design and for other devices, so I utilized my experience in Flash Professional (and ActionScript 3) and graphic user interface (GUI) to create this application. I used a qualitative method by giving users a questionnaire and collecting their feedback on the application’s usability. I then distributed or shared the application on various social networking sites, i.e. Facebook and Twitter.

The interactive application contains sinographs, ancient seals (ancient Chinese characters), traditional characters and English meanings. It is in flashcard form. And, finally, a quiz (in drag & drop puzzle form) is provided to test users' knowledge of sinographs after they learn or memorize the assigned characters. Users have to combine pieces to form sinographs to match the English meaning.
## Outline

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td><em>Problem Statement</em></td>
<td>2</td>
</tr>
<tr>
<td>Process</td>
<td>5</td>
</tr>
<tr>
<td><em>Plan</em></td>
<td>5</td>
</tr>
<tr>
<td><em>Experiment</em></td>
<td>5</td>
</tr>
<tr>
<td><em>Research</em></td>
<td>6</td>
</tr>
<tr>
<td><em>Contents</em></td>
<td>6</td>
</tr>
<tr>
<td><em>Interface Design</em></td>
<td>7</td>
</tr>
<tr>
<td><em>Technical Issues</em></td>
<td>11</td>
</tr>
<tr>
<td><em>Troubleshooting</em></td>
<td>12</td>
</tr>
<tr>
<td>Summary</td>
<td>13</td>
</tr>
<tr>
<td><em>Qualitative Method &amp; Audience Feedback</em></td>
<td>13</td>
</tr>
<tr>
<td>Conclusion</td>
<td>14</td>
</tr>
<tr>
<td>Appendix</td>
<td>15</td>
</tr>
<tr>
<td><em>ActionScript 3 Coding Examples</em></td>
<td>15</td>
</tr>
<tr>
<td><em>Audience Feedbacks</em></td>
<td>20</td>
</tr>
<tr>
<td>Bibliography</td>
<td>23</td>
</tr>
</tbody>
</table>
Introduction

Problem Statement

In the same way that people rely on keyboards on laptops, or handheld phones, people also rely on Roman letters as an input method to enter sinographs; in addition they rarely write anymore and often forget how to memorize proper characters.

To overcome these issues, people have the ability to use Flash interaction on their iPhone/iPods/iPads as a learning tool. I designed and developed an interactive application to assist Chinese learners to understand the meanings of and memorize characters and their graphic features. The application provides flashcards of the most popular and regularly utilized characters.

In most other applications, people learn to write strokes (lines, squares and dots) to make a character without recognizing the actual meaning of the character’s radical. In this application, individuals can form a character by combining radicals (as done in the drag & drop puzzle), all the while recognizing these radicals as the origins of the characters and their meanings.

It is my hope that this prototype project will provide Computer Graphics Design developers with the knowledge to apply or convert Flash interaction for iPhone/iPod/iPad applications.
Process

Plan

To create a prototype project, it’s necessary to craft a comprehensive plan for the interaction design, including all the contents. The overall purpose of the plan is to ensure that users enjoy learning these characters without becoming overwhelmed. Here is my take on this:

1. Users will learn without becoming overwhelmed if provided with only a small number of characters.
2. Users will stay interested in learning by utilizing an attractive and fun-looking interface.
3. Users will enjoy playing the “game” by using the drag & drop puzzle with scores and feedback.

The plan did not, however, remain static. It evolved as I experimented, researched and received feedback on the project. For example, I originally planned to come up with 100 characters, but after I researched the majority of learners’ learning methods, I reduced this to 50 characters so as not to stress users overly much.

Experiment

Before I began the project, I experimented with what I learned from courses I took during my graduate years, as well as with a few new things I picked up from online tutorials and discussion forums. I brushed up on my ActionScript 3.0 skills and reviewed some of my previous Flash projects, especially those related to creating quizzes and drag & drop interactions. I tested and applied a few small Flash drafts on my own iPhone. Sometimes a friend or classmate would make a suggestion or offer an alternative method in order to optimize Flash contents for iPhone. For example, I attempted to load Flash content on my iPhone and it took a great deal of time. It was
Research

To create the contents for the project, I researched frequently-used Chinese characters in the Chinese and Japanese languages. I read blogs and forums where learners discussed their learning methods for these characters. For example, one learner learned 80 characters per two weeks and another learned 10 characters per day. Also, I analyzed each character’s etymological origin so that I could provide an explanation for each character.

As my research into sinographs progressed, I was faced with various complications, i.e. classifications. I attempted to simplify down a bit to some easy-to-understand information for users. I divided 50 characters into 20 basic radicals (or pictograms that imitate objects they represent), five crown elements (one radical above another radical), five left-side elements (one radical being next to another radical’s left), and five dangle elements (one radical hanging over another radical). For the project, there will be primitive forms next to each to help users to see the similarities between the characters and the objects/meanings the characters represent.

Contents

Following this research, I decided upon the content of the project. I divided the contents into six sections. For the first section, I wrote the introduction and history of sinographs and provided further understanding of forms and radicals. For the second and third sections, I put in ten basic radicals in each section. The ten 10 radicals in the second section have fewer than four strokes, while the other ten radicals in the third section have more than four strokes. For both sections, I created a quiz wherein users are able to choose one of four flashcards to match to the English meaning. The fourth section contains crown element radicals; the fifth section has left-sided element radicals, and the sixth section for contains dangle radicals. The quizzes in those sections are different.
from the quizzes in the second and third sections as they require users to put two of four flashcards together to match the meaning and they must be in the correct right order.

**Interface Design**

As I mentioned earlier, my hope is to keep users attracted to the interaction based on its interface design. In my thesis proposal, I crafted a very rough draft for the interface design (Figure 1) based on iOS interface design (Mail, Contacts and Calendar), but I never adopted it; simply put, it didn’t meet the plan expectations nor did it show any creativeness or originality.

![Figure 1. Rough draft of interface design](image)

I continued brainstorming and experimenting with a few interface designs on Adobe Photoshop and Illustrator. I then came up with a combination of the “ancient” feeling and “minimalist” design (Figure 2). After testing this successfully on the iPhone, I found it “detached” from the feeling of “studying.” The color choices weren’t pleasing and were too dark for the concept of “fun” learning.
Figure 2. Earliest and first interface design

After various one-on-one meetings with the thesis committees and discussions with peers, several individuals agreed that the interface design looked “detached.” I went back to Adobe Photoshop to experiment again.

Figure 3. Composition cards in the Tale of Genji Museum

I recalled the Chinese character composition cards in the Tale of Genji Museum in Kyoto, Japan (Figure 3). I liked them and compared them to the students studying flashcards. I thought that utilizing the students’ supplies as an interface theme was creative. From this idea, I created the stack of flashcards on a wooden desk (Figure 4).

After the one-on-one meetings with the thesis committees and based on their feedback, I changed some of the fonts and a few areas of the interface design; for example, I added texture to the flashcards and changed few words.

In the next step, I determined how to demonstrate just one Chinese character to the users for them to learn and included the radical, an explanation, its primitive form and its traditional form (if it existed) on one screen. I did not want to force users to read too much on one small screen, e.g., the iPhone screen size. I designed the character demonstration following interface design as examples in the table bottom.
Rough draft of character demo (Figure 1).

Early and first interface design (Figure 2) of character demo. You could tap “radical” to show radical compositions and “explanation” to show the details of character.

Semi-final interface design (Figure 4) and final interface design (Figure 5) of character demo. A user would note “Tap the card…” at the bottom of flashcard. Once it’s tapped, it would “flip” (not in animation) to show other forms such as the primitive form and traditional form. Three icons, “previous,” “menu” and “next,” were provided for users’ convenience.

Finally, I designed the interface for the quiz wherein there are four small flashcards and a user must pick and drag one or two flashcards and drop them upon target box to match the character’s English meaning. I reduced the eight radical cards in the early design to four (semi-final design) to lessen the “crowded” feeling. The quizzes from the two interface designs can be compared in Figure 6.
Technical Issues

The greatest challenge and recurring nightmare I had to face was the ActionScript 3 coding in the drag & drop quiz. Two issues I faced included problems related to randomly arranging the small flashcards when someone went on to the next question and how to allow two answers to make up one answer. For the random arrangement, it required me to research previous Flash projects from other courses in order to find and review similar codes. I often Googled similar code errors and looked for alternative codes. I finally determined that I would have to create an array and shuffle for the small flashcards’ x-axis (Figure 7).

```javascript
var aLocal:Array = new Array("48.50","125.50","202.50","280.50");

function shuffle(aLocal,bLocal):Number
{
    var num:Number = Math.round(Math.random()*5) - 1;
    return num;
}

function reArrange()
{
    var bLocal:Array = aLocal.sort(shuffle);
    radical_1.x = bLocal[0];
    radical_2.x = bLocal[1];
    radical_3.x = bLocal[2];
    radical_4.x = bLocal[3];
    radical_1.x = bLocal[0];
    radical_2.x = bLocal[1];
    radical_3.x = bLocal[2];
    radical_4.x = bLocal[3];
}
```

Figure 7. Four different x-axis numbers in array and the use of shuffle as sort.

To allow two answers to come together to make one answer, I was able to write code to have one card for one answer. I struggled to come up with code to have two cards for
one answer. I turned to one of my thesis committees for a solution. I was provided with an idea/small code example to get me started. I realized that I would need to write two arrays for two answers. For example, I wrote first array for 10 radicals (Line 1779). Then I wrote second array (Line 1782) to include five radicals from first array, and then wrote third array (Line 1783) to include other five radicals from first array -- but they had to exactly match with those in the second array in the exact same order. Here is the example (Figure 8).

```javascript
var Radicals1:Array = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '10'];
var RadicalsTypes:Array = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
var QuizAnswersFirstRadical3:Array = [1, 3, 5, 7, 9];
var QuizAnswersSecondRadical3:Array = [2, 4, 6, 8, 10];
var IncorrectRadicals3:Array = [6, 2, 3, 4, 1];
var IncorrectRadicals5:Array = [8, 4, 10, 3, 2];
var Questions3:Array = ['Divide', 'School', 'Meet', 'Home', 'Need', 'Require'];
var UserAnswers3:Array = ['分', '格', '家', '家', '求'];
var QuestionNumbers3:Number = -1;

function QuizQuestion3(event:MouseEvent):void
{
    removeChild(this_mc3);
    reArrangement();
    QuestionNumbers3 = -1;
    QuestionNumbers3++;
    match_mc.matchFile_txt_text = Questions3[QuestionNumbers3];
    radical_1.radical_txt_text = Radicals3[QuizAnswersFirstRadical3[QuestionNumbers3]];
    radical_2.radical_txt_text = Radicals3[QuizAnswersSecondRadical3[QuestionNumbers3]];
    radical_3.radical_txt_text = Radicals3[IncorrectRadicals3[QuestionNumbers3]];
    radical_4.radical_txt_text = Radicals3[IncorrectRadicals5[QuestionNumbers3]];
    cand_mc.finalCharacter_txt_text = UserAnswers3[QuestionNumbers3];
    correct_mc3.radical_txt_text = Radicals3[QuizAnswersFirstRadical3[QuestionNumbers3]];
    correct_mc3.radical_txt_text = Radicals3[QuizAnswersSecondRadical3[QuestionNumbers3]];}
```

Figure 8. Two arrays borrow radicals (Line 1779) and arrange numbers, parallel to each others (Line 1782-1783)

Troubleshooting

One major lesson I’ve learned throughout this process is to continuously maintain communication with my thesis committees and classmates who are well-versed in interface design and ActionScript 3. Various discussions with them and feedback from them helped me a great deal. For example, their feedback caused me to completely change my initial interface design. The final interface design (Figure 5) looks much better than the semi-final interface design (Figure 4). I could simply have asked my classmates for some coding assistance instead of sitting through the many test code errors that cost me many days and delays. I found few solutions by Googling errors and rewrote codes all over again – basically recreating the wheel.
In order to receive audience feedback, I crafted a webpage (http://www.paulguo.com/test_app) containing the Flash project in pseudo iPhone and a form contact requesting background briefing and feedback. Originally, I was going to share the Flash project in iOS app with people who have an iPhone or an iPod. The results of the survey indicated problems with testing the app on their iPhones/iPods, even though it worked well on my own iPhone. This feedback was very helpful and I decided to keep this aspect in my plan.

The majority of the feedback I received was with regard to technical issues, especially about one misspelled word, “harmony,” and some parts of the quiz that weren’t functional. One user suggested several things, i.e. a timer, randomized questions and a final section with mixed questions from all the previous sections. There was positive feedback, too; some commented on how nice the interface design was and another said, “it is easy to understand and easy to learn.”
Conclusion

My original plan was to teach users how to write through animation and how to read in Mandarin, Japanese and Korean through audio. As I read and learned more about applying Flash content to the iPhone as an app, and received clarification from the thesis committees and feedback from peers, the plan went through much iteration, i.e. animation and audio were omitted. In the end, the project that was based on the changed plan went well, despite minor technical issues. I am not only satisfied with my project, but am satisfied with the extent of the knowledge I gained throughout process.

I believe that knowledge gained from any process is very important. With my broken Chinese language background, I am constantly trying to research and discover new information regarding Chinese characters. This project is also a learning experience for me; although at first I was weaker in interface design and ActionScript 3, I learned to appreciate the feedback from the thesis committees on my interface design. Without this feedback, I would have received less positive feedback from the end-users. ActionScript 3 coding was the most challenging aspect of the process and more difficult than I had anticipated. Working through this process, however, affected my ability to code in a positive way.

The project, including the feedback I received from the end users, however, left me with some unanswered questions. Could I have used a different programming application to create this project for iPhone instead of Adobe Flash? Should I have created web interaction instead of phone interaction? To answer the first question, I realized I should have utilized something simpler than ActionScript 3 and Adobe Flash for this project. As for the second question, when I saw the results of using a webpage to allow users to try out the pseudo iPhone app, I felt that I should have done web interaction, similarly to the first question. But I do not regret at all how much time I spent in attempt to gain more knowledge.
Appendix

ActionScript 3 Coding Examples

Random arrangement
This coding example is to create and arrange radical cards’ x-axis numbers randomly in drag & drop quiz.

```
    sradical_1 = new cardRadical ;
    sradical_1.y = 336.50;
    sradical_1.radical_txt.defaultTextFormat = myFormat4;
    sradical_1.radical_txt.embedFonts = true;

    sradical_2 = new cardRadical ;
    sradical_2.y = 336.50;
    sradical_2.radical_txt.defaultTextFormat = myFormat4;
    sradical_2.radical_txt.embedFonts = true;

    sradical_3 = new cardRadical ;
    sradical_3.y = 336.50;
    sradical_3.radical_txt.defaultTextFormat = myFormat4;
    sradical_3.radical_txt.embedFonts = true;

    sradical_4 = new cardRadical ;
    sradical_4.y = 336.50;
    sradical_4.radical_txt.defaultTextFormat = myFormat4;
    sradical_4.radical_txt.embedFonts = true;

    for (var n:Number = 1; n <= 4; n++)
    { 
        var cur:cardRadical = this["radical_" + n];

        cur.originalX = cur.x;
        cur.originalY = cur.y;
        cur.addEventListener(MouseEvent.MOUSE_DOWN,dragPiece);
    }

    for (var s:Number = 1; s <= 4; s++)
    {
        var scur:cardRadical = this["sradical_" + s];
        scur.originalX = scur.x;
        scur.originalY = scur.y;
        scur.addEventListener(MouseEvent.MOUSE_DOWN,dragPieceSingle);
    }
```
var aLocal:Array = new Array("48.50","125.50","202.50","280.50");

function shuffle(aLocal,bLocal):Number
{
    var num:Number = Math.round(Math.random() * 5) - 1;
    return num;
}

function reArrange()
{
    var bLocal:Array = aLocal.sort(shuffle);
sradical_1.x = bLocal[0];
sradical_2.x = bLocal[1];
sradical_3.x = bLocal[2];
sradical_4.x = bLocal[3];
}

Basic radical makes one answer to one question

This coding example is to create basic radical for each card; to answer with one card.

var Radicals2:Array = ["","方","不","以","中","心","主","用","自","来","面"];
var RadicalType2:Array = [0,1,1,1,1,1,1,1,1,1,1];
var Quiz1AnswersFirstRadical2:Array = [1,2,3,4,5,6,7,8,9,10];
var IncorrectRadicalx2:Array = [2,3,5,1,8,10,2,9,5,2];
var IncorrectRadical3:Array = [5,4,6,10,1,1,4,7,3,1];
var IncorrectRadical4:Array = [3,8,9,7,4,9,3,6,2,6];

var Questions2:Array = ["Way, Direction","Negative, Bad","By means of","Middle","Heart, Mind","Master","Use","I, Self","Come","Face"];
var UserAnswers2:Array = ["","方","不","以","中","心","主","用","自","来","面"];

var QuestionNumber2:Number = -1;

function QuizQuestion2(event:MouseEvent):void
{
    removeChild(ready_mc2);
    reArrange();
    QuestionNumber2 = -1;
    QuestionNumber2++;
    //QNum.text=QuestionNumber;
    match_mc.match_txt.text = Questions2[QuestionNumber2];

    sradical_1.radical_txt.text = Radicals2[Quiz1AnswersFirstRadical2[QuestionNumber2]];
    sradical_2.radical_txt.text = Radicals2[IncorrectRadicalx2[QuestionNumber2]];
    sradical_3.radical_txt.text = Radicals2[IncorrectRadical3[QuestionNumber2]];
}
srادیکال_4.رادیکال_txt.text =
    Radicals2[IncorrectRadical4[QuestionNumber2]];
card_mc.finalCharacter.text = UserAnswers2[QuestionNumber2];
correct_mc.radical_txt.text =
    UserAnswers2[QuestionNumber2];
}

function NextQuestion2(event:MouseEvent):void
{
    QuestionNumber2++;
    reArrange();
    //QNum.text=QuestionNumber;
    match_mc.match_txt.text = Questions2[QuestionNumber2];

    srادیکال_1.رادیکال_txt.text =
        Radicals2[Quiz1AnswersFirstRadical2[QuestionNumber2]];
    srادیکال_2.رادیکال_txt.text =
        Radicals2[IncorrectRadicalx2[QuestionNumber2]];
    srادیکال_3.радیکال_txt.text =
        Radicals2[IncorrectRadیکال3[QuestionNumber2]];
    srادیکال_4.رادیکال_txt.text =
        Radicals2[IncorrectRadیکال4[QuestionNumber2]];
card_mc.finalCharacter.text = UserAnswers2[QuestionNumber2];
correct_mc.radical_txt.text = UserAnswers2[QuestionNumber2];
card_mc.visible = false;
good_mc.visible = false;
bad_mc.visible = false;
correct_mc.visible = false;
srادیکال_1.visible = true;
sradical_2.y = 336.50;
sradical_3.y = 336.50;
sradical_4.y = 336.50;

if (QuestionNumber2 == Questions2.length)
{
    if (myScore == 100)
    {
        note_mc.visible = true;
        note_mc.note_txt.text = myScore + " points! Well done!";
    }
    else if (myScore >= 80)
    {
        note_mc.visible = true;
        note_mc.note_txt.text = myScore + " points! Not bad!";
    }
    else if (myScore < 79)
    {

note_mc2.visible = true;
note_mc2.note_txt.text = myScore + " points? Try again.";
}
}
}

**Two radicals make one answer to one question**

This coding example is to create two answer cards for one question.

```javascript
var Radicals3:Array = ["","八","刀","人","云","字","家","西","女"];
var RadicalType3:Array = [0,2,2,2,2,2,2,2,2,2];

var Quiz1AnswersFirstRadical3:Array = [1,3,5,7,9];
var Quiz1AnswersSecondRadical3:Array = [2,4,6,8,10];
var IncorrectRadical5:Array = [6,2,3,4,1];
var IncorrectRadical6:Array = [8,9,10,5,2];

var Questions3:Array = ["Divide","School","Meet","Home","Need, Require"];
var UserAnswers3:Array = ["分","学","会","家","要"];

var QuestionNumber3:Number = -1;

function QuizQuestion3(event:MouseEvent):void
{
    removeChild(ready_mc3);
    reArrange();
    QuestionNumber3 = -1;
    QuestionNumber3++;
    match_mc.match_txt.text = Questions3[QuestionNumber3];

    radical_1.radical_txt.text =
        Radicals3[Quiz1AnswersFirstRadical3[QuestionNumber3]];
    radical_2.radical_txt.text =
        Radicals3[Quiz1AnswersSecondRadical3[QuestionNumber3]];
    radical_3.radical_txt.text =
        Radicals3[IncorrectRadical5[QuestionNumber3]];
    radical_4.radical_txt.text =
        Radicals3[IncorrectRadical6[QuestionNumber3]];

    card_mc.finalCharacter.text = UserAnswers3[QuestionNumber3];
    correct_mc2.radical_txt1.text =
        Radicals3[Quiz1AnswersFirstRadical3[QuestionNumber3]];
    correct_mc2.radical_txt2.text =
        Radicals3[Quiz1AnswersSecondRadical3[QuestionNumber3]];
}
```
function NextQuestion3(event:MouseEvent):void
{
    reArrange();
    QuestionNumber3++;
    match_mc.match_txt.text = Questions3[QuestionNumber3];

    radical_1.radical_txt.text =
        Radicals3[Quiz1AnswersFirstRadical3[QuestionNumber3]];
    radical_2.radical_txt.text =
        Radicals3[Quiz1AnswersSecondRadical3[QuestionNumber3]];
    radical_3.radical_txt.text =
        Radicals3[IncorrectRadical5[QuestionNumber3]];
    radical_4.radical_txt.text =
        Radicals3[IncorrectRadical6[QuestionNumber3]];

    card_mc.finalCharacter.text = UserAnswers3[QuestionNumber3];
    correct_mc2.radical_txt1.text =
        Radicals3[Quiz1AnswersFirstRadical3[QuestionNumber3]];
    correct_mc2.radical_txt2.text =
        Radicals3[Quiz1AnswersSecondRadical3[QuestionNumber3]];
    card_mc.visible = false;
    good_mc.visible = false;
    bad_mc.visible = false;
    correct_mc2.visible = false;
    radical_1.visible = true;
    radical_2.visible = true;
    radical_1.y = 336.50;
    radical_2.y = 336.50;
    radical_3.y = 336.50;
    radical_4.y = 336.50;
    target_mc.y = 168;
    target_mc2.y = -112;

    if (QuestionNumber3 == Questions3.length)
    {
        if (myScore == 100)
        {
            note_mc.visible = true;
            note_mc.note_txt.text = myScore + " points! Well done!";
        }
        else if (myScore >= 80)
        {
            note_mc.visible = true;
            note_mc.note_txt.text = myScore + " points! Not bad!";
        }
        else if (myScore < 79)
        {
            ...
note_mc2.visible = true;
note_mc2.note_txt.text = myScore + " points? Try again."
}
}

**Audience Feedbacks**

I received feedbacks from audience through the webpage where they participated and gave feedbacks, in May 2011.

**Name:** Christopher Leise  
**Age:** 25  
**Educational Level:** Bachelor's Degree  
**Brief:** I will pursue a job position as an accountant. As the world is growing globalized, I recognize the importance of learning new languages for my career as well as my personal life. I took two Japanese classes, which were required for my degree. I feel there is a need for help in learning over 2000 kanji characters.  
**Feedbacks:**  
There is one little problem. Those two boxes seem to delete my texts if I click in this box again.

**Name:** Dylan Nulph  
**Age:** 21  
**Educational Level:** College  
**Brief:** I don't know Chinese or Japanese but I do want to learn Japanese someday.  
**Feedbacks:**  
I really like this. I had so much fun learning simple words then taking the quiz to see if I can remember. Keep the good work up!

**Name:** Yanick Létourneau  
**Age:** 20  
**Educational Level:** College  
**Brief:** Self-learned Han Characters from Japanese (Kanji).  
**Feedbacks:**
"House" cannot be clicked on.
Typo on "Harmony" in quiz.
Maybe clarify what is the order? (Top to bottom in Crowns, Left to Right in Left-sided, Outide to Inside in Dangle)"What is Sinograph?" should be written as "What is a Sinograph?" or "What are Sinographs?"

Name: Dae-Kun Kim
Age: 26
Educational Level: Graduate School (MS degree)
Brief: I took a Chinese class when I was at Korea, and still am exposed to little bit of Chinese here and there. I had a roommate who also taught me little bit of Chinese as well.
Feedbacks:
might wanna change the header font when starting the unit, or definitions of hanja. while young people can read without problems, consider old people too.

Name: Young Jae Kim
Age: 24
Educational Level: University Graduate
Brief: Major in East Asian Studies and completed multiple language courses in their native countries.
Feedbacks:
- when you tap for next question in quiz, you can highlight the character
- timer in quiz mode
- high score based on time and score (faster you complete it with a higher score = better)
- questions randomized in quiz
- indicator stating how many strikes you have before having to redo the test
- a final review section that randomly mixes questions from all sections
- maybe when a new character is shown, it will display the order of brush strokes so users will be able to learn how to properly write these characters
Name: Olivia
Age: 27
Educational Level: College Graduate
Brief: Fluent in conversational Cantonese but not reading; no classes taken. Took an elementary class in Japanese.
Feedbacks:
"Harmony" is misspelled in the quiz portion. Overall, a nice app that offers a good breakdown of characters. Easy to understand and easy to learn.

Name: Chi Hieu Ha
Age: 23
Educational Level: Student
Brief: Absolute newbie
Feedbacks:
Sometimes the screen doesn't show any word to choose, or the meaning of the word doesn't appear on the quiz.

Name: Raphaël
Age: 22
Educational Level: College/cégep
Brief: Finished high school, started college. I have some basis in Japanese which I started to study two years ago. I also know a few words in Chinese mandarin.
Feedbacks:
Interface is great, Sinographs is simple to use and it's good for someone who begins to learn sinographs. Only thing I noticed though is that when you make a mistake (when the incorrect message appears), if you go back to the menu and do the same exercise over again, once you get to the test, the old incorrect message is still there with the ready to go message. It's like this with every sinographs chapters with the test. Maybe you could fix this. Overall, very good app!
Bibliography


Evans, Judith. (2010). *Wired youth forget how to write in China and Japan*. Available: [http://www.google.com/hostednews/afp/article/ALeqM5jz3FEk2KJw3NEUyDhbMITQO0iIOw](http://www.google.com/hostednews/afp/article/ALeqM5jz3FEk2KJw3NEUyDhbMITQO0iIOw). Last accessed 26 October 2010