

VG News!



Richie Thompson
Final Report
December 8, 2009

Table of Contents

Cover Page.....	1
Table of Contents.....	2
Application Development.....	3
Project Description.....	3
Motivation for Choice of this Project.....	4
Project Resources.....	4
Software Requirement Specifications.....	5
Vision and Scope	7
System Design and Architecture	8
Implementation	9
About the Author	9
Known Bugs and Other Issues.....	10
Preliminary Test Plan and Preliminary Test Cases	10
Software Demo(s)	11
Final Codebase and Documentation Access	12
Executive Reports	13

Application Development

Project Description

-Revised September 22, 2009 made less formal

This project will involve the designing and implementation of an iPhone application that will allow the user to receive updates from a select few gaming websites that update regularly. When the application is loaded this app will immediately go to the website online and see if any new headlines have been posted since the time the application was last loaded, if any new headlines are available it will notify the user so that they can select which site they would like to load the headlines from. When the user selects which headline to read the app will load that headline in the app for the user without having to load Safari. This app will assist gamers that have an iPhone who like to keep up to date with websites such as Joystiq, Gamingbits.com, and qj.net without having to load safari and browse to each site or boot up their computer and browse to each site individually. This can be a very time consuming and tedious process. It will be much simpler to have a single application get the updates for you and load them on the screen. There are no other applications for the iPhone/iTouch exists that is similar to this concept.

To start the focus of the application will be to receive updates from joystiq.com, as it is a popular gaming news website that many gamers use to stay up to date. Once joystiq is working the application will grow into incorporating gamingbits.com and qj.net. Other websites other than video game related sites (i.e. comingsoon.net) could be added to this application but this will depend on getting the first three websites working perfectly first. If development of the application continues and a development moves to an Intel processor based MAC machine then development can be done to make the application compatible with iPhone firmware 3.0 which would allow the application to update the headlines itself even if the application is not open. But since the SDK software version 3.0 is not compatible on a PowerPC MAC that development is not currently possible. Using the push notification, much like email does to alert the user when a new email is received, would be useful in that the user would always know when a headline is published even if they are not running the application; the only drawback would be a possible reduction of battery life.

In building this application the biggest issue will presumably be grabbing the headlines from the websites and loading the content on the iPhone's screen for the user. For this to work the application will have to be able to:

- Determine the time a headline is published
- Compare that time to the last time the application was loaded
- Gather the article's content and print in on screen

These three factors seem easy in concept, but in reality it could take time to work with SDK to have it access the internet and grab the articles. The project will be finished when the application can be loaded, it accesses the internet to find new headlines for the website, it notifies the user of any new updates to the site, and finally it prints the article on screen for the user. The first aim is for all this to work with joystiq.com. The final product of the application will be able to at least grab new headlines from joystiq.com, any other websites will be added if time allows

Motivation for choice of this project

When I first heard that the Senior Seminar course was taking a Pilot Project direction and we would be allowed to work on a project of our choosing I immediately started to brainstorm several ideas that I could work on. The iPhone application was the first idea that came to mind, and I tried several times to come up with several other options, but I could not think of any that would match my desire to work on this particular idea. I feel this idea will help me to learn a new coding language, as well as gain some experience working with an operating system that I am not already comfortable with. So, with the career field I will be going into, any experience I can manage to obtain working with different operating systems and devices will assist me greatly. This application does have a very particular audience when it will first be finished and that is anyone who likes video games. As I mentioned this application will be designed to keep the user informed about any new headlines related to video games on the select websites. In the future when I start working on including more than just gaming websites, and the audience will increase to anyone who would like to be kept up to date with whatever sites the application is compatible with.

Project Resources

-This section has been reorganized and updated so it is easier to read and manage September 22, 2009

-Edited the bullets to distinguish resources with their own bullets and edited the textbook I am using September 28, 2009

-Added two new sites that have been helpful or useful in the development October 20, 2009

- Macintosh Laptop with 10.5 OS installed – supplied by IS&S
- iPhone SDK 2.2.1 – iPhone application development software for Macs
- Microsoft Paint – for creating the application icon
- Mark, Dave and Jeff LaMarche. Beginning iPhone Development: Exploring the iPhone SDK. New York: Springer-Verlag New York, Inc., 2009.
- <http://icodeblog.com/2008/07/26/iphone-programming-tutorial-hello-world/> - learned a few basics in editing text of the columns in a navigation based application
- <http://www.xprogress.com/post-33-how-to-add-an-icon-to-your-iphone-application/> - learned how to create an icon for the application
- <http://apokalypsesoftware.com/blog/2009/03/23/130/> -for the editing of the iPhone software to work on a PowerPC MAC.
- <http://theappleblog.com/2008/08/04/tutorial-build-a-simple-rss-reader-for-iphone/> - found the code to parse an xml document
- <http://feeds.feedburner.com> – converts the headlines for the sites I will be using into a rss feed
- Websites used in application:
 - <http://www.joystiq.com>
 - <http://www.gamingbits.com>
 - <http://www.qj.net>

Software Requirement Specifications:

-New material as of September 22, 2009

-Edited bullets to distinguish requirements from resources September 26, 2009

-Edited Functional Requirement F-02, F-04, and added F-06 October 13, 2009

-Edited Functional Requirement F-03

-Edited Functional Requirement F-06 November 10, 2009

-Added a new Functional Requirement put in place of F-03, previous requirements bumped down November 16, 2009

-Added a new Functional Requirement put in place of F-04, previous requirements bumped down December 1, 2009

-Reorganized so the 'High' priority functional requirements are first, then 'Low', and finally 'If Time Permits' December 7, 2009

- Functional:
 - Number: F-01
 - The application must present a user interface that is simple and easy to use
 - Will be determined working once the application is loaded and anyone can easily navigate through the menus
 - Dependency: None
 - Priority: High
 - Requirement Revision History: added to the list set as high priority, if a simple user interface cannot be created for the application than no one will want to use the application for the phone and it will fail
 - Number: F-02
 - Software must be able to access the internet and retrieve the latest headlines for www.joystiq.com and present the headlines to the user in a table (10/13)
 - Will be working when the application displays the headlines for each site in a simple table for the user
 - Dependency: F-01
 - Priority: High
 - Requirement Revision History: added to the list and set as high priority, this function will not be started until the user interface is working, without an interface having the application access the internet is pointless
October 13, 2009 – limited the number of sites to just one, initially only one site will be worked with to retrieve headlines for
 - Number: F-03
 - Software will be able to open the article for the associated headline in the Safari application installed on the iPhone
 - Will be working when a headline can be selected and the article will open in Safari for the user to read
 - Dependency: F-02
 - Priority: High

- Requirement Revision History: added to the list of requirements and set the priority level to High. This requirement needs the application to grab the headlines before the matching URL can be opened in Safari.
- Number: F-04 (10/13)
 - Application will load headlines for multiple sites for the user to read
 - Will be determined when application loads and a few sites are presented on the main screen and each can be navigated to read articles
 - Dependency: F-01
 - Priority: High (11/10)
 - Requirement Revision History: October 13, 2009, added to the list of requirements, the application will load one site initially and if time permits more sites will be included into the application
- Number: F-05 (12/01)
 - Application will rotate the display when the iPhone is rotated
 - Will be working when the views of the applications rotate properly when the iPhone is rotated
 - Dependency: None
 - Priority: Low
 - Requirement Revision History: added to the list of requirements with a Low priority.
- Number: F-06
 - Software must be able to display the text on the iPhone screen in a readable fashion for the user
 - Will be determined working when the entire headline is able to be read on the iPhone screen
 - Dependency: F-02
 - Priority: If Time Permits (10/31)
 - Requirement Revision History: added to the list and set as high priority, function will allow the application to be more user friendly without the need to load the Safari browser, application needs to be able to access the internet first
- Number: F-07
 - Software will inform the user if any new headlines have posted since the last time the application was loaded
 - Will be determined working when after the application is loaded the websites are labeled with a number showing how many new headlines have been posted
 - Dependency: None
 - Priority: If Time Permits (10/13)
 - Requirement Revision History: added to the list and set as middle priority, this function is not completely necessary but will allow the user to know if a new article is available to read
October 13, 2009 – priority reset to if time permits, the refresh button will not be worked on until the basic function of the application is finished
- Number: F-08
 - Software will offer the user a refresh option so the user can refresh the application to see if any new headlines have posted while the application was running
 - Will be determined working when a refresh button appears on screen that, when pressed, will update the application with any new headlines

- Dependency: None
 - Priority: If Time Permits
 - Requirement Revision History: added to the list of requirements, if time allows for this feature the user will not have to close the application and relaunch it just to update the headlines that are posted, the application will have a button that when pressed will access the internet and look for any new headlines
- Non-Functional:
 - Number: NF-01
 - Software must run in the iPhone operating System
 - With the iPhone SDK software providing an iPhone simulator the application will work in the operating system as long as the simulator runs the application without crashing
 - Dependency: None
 - Priority: High
 - Requirement Revision History: added to the list of requirements, this is completely necessary to the success of this application. Since the application is meant to work on the iPhone it will need to run in the iPhone operating System

Vision and Scope:

-New material as of September 26, 2009

-Edited scope portion October 31, 2009

In today's society countless gamers log onto websites such as Joystiq, or GamingBits with the intent on reading the latest headlines from gaming developers around the world. What this project will accomplish is present gamers who own an iPhone or iPod Touch with a means to easily stay up to date with all the new headlines in the gaming community. They will no longer need a computer with internet access to be able to read articles on these respective sites. All they will have to do is take out their device and load the application, and as long as they are in range of a cell phone tower they will be able to access the articles from anywhere. Gamers will find keeping up with headlines with each of these sites simple and easy, and they will feel like they are more in sync with the gaming world.

This application will load the latest headlines that each site has published, and if the user selects one of the headlines the application will open the URL of that headline in Safari. The application will not inform the user if a headline is new, but it will be able to search the internet again for refresh itself at the user's command. This is one of the few drawbacks this application will have, another is that if an article has a video embedded in it that video will not be able to be played, to view the videos the user will need access to a laptop. This application will also allow the user to view old articles each site has published, though there will be some limitations to how many articles the application will be able to provide the user.

System Design and Architecture

-Edited on October 27, 2009 to show the new architecture setup

Once the application is completed it will present the user with a table that lists Joystiq, when the user selects Joystiq the application transfers control to a second table that lists headlines that are available to read on joystiq.com for the user to read. If the user selects an article to read then that article is printed on screen for the user to read.

- Software Components:
 - ◆ **VG News 0 5AppDelegate(.h/.m)** – declares the window and navigation controllers for the application to use
 - ◆ **RootViewController(.h/.m)** – a controller class for the main view of the application, this controls what is displayed in the title bar and the first table the user views, contains navigation information so the user can navigate back to the root view after selecting Joystiq or QJ
 - ◆ **SecondLevelViewController(.h/.m)** – a controller class for the second view of the application, once the user has selected a site the RootView class passes control over to the SecondLevelView which prints the articles on screen for the user
 - ◆ **MainWindow.xib** – this file controls the main window where the tables and articles are displayed, basically this file keeps the title bar and table aligned properly
 - ◆ **SecondLevel.xib** – file to control the table of headlines and how they will be printed on screen

When program is loaded:

This is the first step the application loads, the VG_News__2_0AppDelegate loads the Main Window of the application which displays the RootViewController showing the title bar and table on screen

VG_News__0_5AppDelegate

⇒ **MainWindow.xib**

→ **RootViewController**

When the User selects a row in the table the MainWindow switches to the SecondLevelViewController which contains a table that holds the headlines that were parsed from the internet

VG_News__0_5AppDelegate

⇒ **SecondLevel.xib**

→ **SecondLevelViewController**

There is another step in how the application operates currently. If the user selects a headline to read then the application will open the URL in safari, this method may be the final product currently but that has yet to be determined.

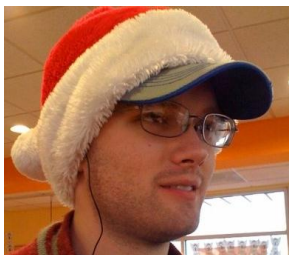
Implementation

- [README.txt](#) (In progress as of October 20, 2009) – a text file that contains the installation instructions and contact information for the program designer
- [Icon.png](#) (In progress as of October 20, 2009) – icon for the application
- [VG News_0_5AppDelegate\(.h/.m\)](#) (In progress as of October 20, 2009) – the header and implementation files for the window and navigation controllers
- [RootViewController\(.h/.m\)](#) (In progress as of October 20, 2009) – a controller class for the main view of the application, this controls what is displayed in the title bar and the first table the user sees, contains navigation information so the user can navigate back to the root view after selecting Joystiq
- [SecondLevelViewController\(.h/.m\)](#) (In progress as of October 20, 2009) – header and implementation file for the second level view of the application, contains the navigation code that will allow the user to navigate to and from the root view, as well as the code to parse the internet site for headlines and store them into the table, contains to code to add the refresh button and what action it will perform when pressed
- [MainWindow.xib](#) (In progress as of October 20, 2009) – this file controls the main window where the tables and articles are displayed, basically this file keeps the title bar and table aligned properly
- [SecondLevel.xib](#) (In progress as of October 20, 2009) – this file controls the view of how the table containing the headlines is displayed on screen

When the application is loaded the MainWindow.xib sets up a window and with the RootViewController telling the Window what to display on screen. When the user selects a site to load the articles from the RootView passes control over to the SecondLevelView, which parses the URL for headlines to print on screen, the headlines are then printed on screen for the user.

About the Author

-New content as of October 27, 2009



Richie Thompson – The Main Man Stan

Born in Fort Wayne Indiana but lived most of his life in Kentucky, you will never meet a bigger University of Kentucky fan than this guy. This may be the first legitimate case where if you cut this guy he will truly bleed blue. A gamer at heart but has a quality that most gamers lack, he has the ability to put the controller down and get more important tasks done first.

Known Bugs and Other Issues – in VG News! 0.5 beta

-New content as of October 31, 2009

- i. Slight delay while headlines are parshed

Preliminary Test Plan and Preliminary Test Cases

-New content as of November 8, 2009

-Updated November 16, 2009, edited the first test case and added a second test case

During testing of VG News! the application will be compiled and loaded into the iPhone simulator, all which is handled by the iPhone SDK. Barring no errors occurring during compiling the application should open automatically in the simulator. Once loaded the sites preprogrammed in will appear on the first view the user sees, from there the user will navigate to each set of headlines to ensure they load, and then to the articles each headline is referencing. Barring no crashes or other unforeseen problems the application will be working as expected.

- **Beta Test Case 01 – BETA00001**

- Description: The first test case will open the application to make sure the user interface is loading on screen and is in fact easy and simple to navigate
- Precondition: The application should be coded enough for it to load into the iPhone simulator with no errors at the time of compiling.
- Steps:
 - Compile the application within the iPhone SDK
 - Run the application within the iPhone simulator
 - Browse through the individual views of the application to ensure ease of use
- Result: At the end of this test the application will present the user with a simple interface they can navigate through. The interface should appear simple with little to no chance of confusion

- **Beta Test Case 02 – BETA00002**

- Description: This test will ensure both websites are appearing on screen. Each website can be navigated to individually with their appropriate headlines being displayed.
- Precondition: The application should have a simple user interface to load the websites into the main view, and have a second view to navigate to where the headlines are displayed
- Steps:
 - Compile the application within the iPhone SDK
 - Run the application in the iPhone simulator
 - Navigate to all websites on the main view separately
 - Ensure the headlines are accurate by checking their validity in a separate browser

- Result: At the end of this test the application will be able to navigate to all sites on the main view individually and display the headlines for each site on screen.
- **Beta Test Case 03 – BETA00003**
 - Description: This test will test the ability of the application to open the article for each headline in Safari for the user to read.
 - Precondition: The application will load the headlines for each site correctly
 - Steps:
 - Compile the application within the iPhone SDK
 - Run the application in the iPhone simulator
 - Navigate to a website of the tester's choosing
 - Select a headline to open the article in Safari
 - Ensure the article is in fact correct by opening the same article in a separate internet browser
 - Run the application again but select a different article to open
 - Test several articles to ensure no problems arise
 - Result: At the end of this test the application will be able to open any article for the headlines it parses in the Safari application with no serious errors
- **Beta Test Case 04 – BETA00004**
 - Description: This test will test the rotating function of the iPhone application
 - Precondition: The application will open
 - Steps:
 - Compile the application within the iPhone SDK
 - Run the application in the iPhone simulator
 - While the application is open use the key commands to rotate the iPhone simulator left and right and make sure the view changes accordingly
 - Result: At the end of this test the application should be able to rotate the display as the iPhone is rotated

Software Demo(s)

-New content as of December 1, 2009

The demo video for my application is a short and simple video that shows all the features of the application. I provide voice over work explaining what the application is doing as I am recording the application while it is being used.

Link: <http://www.youtube.com/watch?v=sFKDo2rtTy8>

Final Codebase and Documentation Access

-New content as of December 8, 2009

The link below navigates to a download link for a RAR file that contains the VG News! application folder, as well as the README.txt file. All that needs to be done is download it and extract it to wherever you please all files are there.

Link: Coming Soon!

Project Title: VG News!
Project Logo:
Richie Thompson
September 10, 2009

For this week I was able to get the software for developing iPhone applications running on the MAC I was provided. This took some research, but after reading what other users had tried to do to fix the problem and finally finding the solution that would work I managed to get it working. This took a good couple of days of trying several different methods I found. Once I settled on the fact the newest software would not work I had to locate a download for the older software which, with the network restrictions, took some time and I almost thought I would not be able to find it. But fortunately I was able to find a download link that worked and made the necessary changes to the installer package and now the program is working fine. The only drawback is I will be unable to install the application on any iPhone running the latest firmware package the application will run in the simulator however.

For the next week I am planning on starting to code the user interface the application will be using. This includes being able to select Joystiq from the first menu and have it navigate to a second page where the headlines will be loaded at. At this page the user will be able to navigate back to the first page where the site could be chosen. I hope to include a place in the user interface that will serve as an "About this App" section, which will explain to the user the current version of the application, and what functionality it has.

Project Title: VG News!



Richie Thompson

September 22, 2009

In the past week I have been unable to start work on the user interface, I was hoping to make a lot more progress with the program but due to unforeseen events I did not have a lot of time to work on the application. I was able to perform more research and found some tutorials and guides that should prove useful in the creation of the application. I did find a tutorial detailing how to create an icon for an iPhone application; following this guide and determining a picture to use I created a simple prototype icon for the application. Creating the icon was fairly quick and easy, and I was glad that I found a textbook that will assist me in creating the application so I do not try and figure out how to code the application without any idea of what I am doing. It took me a few hours to find a downloadable textbook that would assist me in creating the application, and creating the icon took about thirty minutes. So far in this project I have been able to work for about three to four days total on the application, I hope to increase this time in the coming weeks drastically.

This week I will be looking through the textbook so I will get a better idea of how the coding for an iPhone application actually works. I hope to actually start coding the user interface for the application at some point during the week. I am currently planning on creating a navigation based application, this is because the concept I have in mind for presenting the websites to the user resembles the blank template provided by the SDK software. I have been looking at a few different applications on the iPhone application store to get an idea of how I would want the application to be presented and I am confident I have most of the visual concepts ironed out. I will continue to edit the icon for the application as time goes on, but for now this icon will serve as the beta for the application.

Project Title: VG News!



Richie Thompson

September 28, 2009

During the past week I finally started coding the user interface for the application, I feel like I have made a lot more progress with the project than I have in weeks prior. One complication did arise during this past week, while reading over the textbook and learning how coding in SDK and objective-c language worked I did not consider the possibility that the book was referring to SDK 3.0 and that the difference from SDK 3.0 and 2.2.1 would be great enough for a major issue to develop. I was however wrong, after coding for a couple of hours and getting the final pieces in place for the basic table that will hold the names of the websites and headlines, I went to build and run the application in the simulator and several errors were found in the code. I looked at the errors listed and several mentioned references to code that does not exist, it was then that I realized that the code was different in SDK 3.0 and I needed an early textbook that would help with SDK 2.2.1. Fortunately I found the first edition of the same textbook for SDK 3, following this new textbook and editing the code to work in SDK 2.2.1 the application ran in the simulator just fine. To date I have the basic table set up and a title bar with the application title, I have also developed a new icon for the application but I intend to update it in the future.

The total time spent this week on the project is:

- ◆ Initial coding and trying to fix the errors in the application – 2 hours
- ◆ Finding the first edition of the textbook – 1 hour
- ◆ Writing the correct code for the application and debugging – 2 hours
- ◆ Adding a title bar to the table – 30 minutes
- ◆ Developing a new icon – 10 minutes

Total time put into the project to date is 4 days 5 hours and 40 minutes

This week I will be continuing to work on the code for the user interface, I do not think I will have the interface done by the end of the week but I hope to add a tab and navigation bar to the application. Adding a navigation bar to the project is priority currently due to the fact it will add a second table that will be linked to the first table, so the first table will hold the Joystiq site name, and the second table will list all the sites within Joystiq (i.e. Joystiq[All], Joystiq[Playstation], Joystiq[Nintendo], Joystiq[Xbox]). If I can get the navigation bar coded and working I will have one of the biggest portions of the interface out of the way

Project Title: VG News!



Richie Thompson

October 12, 2009

For the previous two weeks I have been able to implement the user interface for the application to a point where I feel like it is complete for the final product. The current interface is extremely basic and simple but I feel like writing and understanding the code to allow the application to grab headlines as well as articles from the internet will be the most challenging and time consuming. If time permits I plan to go back and add more functionality to the user interface, but I feel like this is not as important to complete. Right now the application has half of the final product coded and the more challenging portion is getting ready to get started. Hopefully I will be able to code the application to access the internet in a timely manner that will allow me to go back and add some more features to the application interface.

The total time spent this week on the project is:

- ◆ Coding more of the user interface – 3 hours
- ◆ Editing the code to follow my initial design – 30 minutes

Total time put into the project to date is 4 days 9 hours and 10 minutes

For the coming week I will be working on the portion of the code that will access the internet and print headlines in a table for the user to see. This is the first step in the internet portion and presumably the hardest. As I have yet to code anything for the application to actually access the joystick URL, hopefully I will have the application grabbing headlines and printing them in the table. Once the application grabs the headlines and prints them on screen all that will be left is printing the article on screen for the user to read.

Project Title: VG News!



Richie Thompson

October 19, 2009

For the this past week I have been working on the code for the project to grab the headlines and print on the iPhone screen for the user to read, this as expected is proving to be rather difficult. I was able to find a XML parser that was created for an iPhone application online and using this code and the site feeds.feedburner.com, which fortunately puts the headlines for the sites I will be using into a RSS feed format which the code can then parse and store into an array. This code is giving me problems when trying to print the array on screen. I have tested this code by itself on the root view of the application and it works fine so I know the headlines are being grabbed I just cannot get them to print properly. I spent the majority of my time trying to figure out where the error in my code is at by so far I have been unable to.

The total time spent this week on the project is:

- ◆ Finding any help to code a XML parser for the application – 2 hours
- ◆ Looking over the code to try and determine where the problem is – 2 hours

Total time put into the project to date is 4 days 13 hours and 10 minutes

In the upcoming week I will be continuing my work in getting the array of headlines to print on the iPhone screen. I feel like the solution is right in front of me and I am simply looking over it, but nevertheless I will figure out the problem and fix it. I hope to have this problem figured out by this time next week so I can begin work on printing the headlines on screen. I know there is a way for me to have the article load up in the Safari application, which would add in youtube videos and pictures to the article, but I am still intending on printing the article on screen.

Project Title: VG News!



Richie Thompson

October 27, 2009

For the past week I have been putting more work into figuring out why the headlines were not displaying on screen. After looking over the code without any improvements to its functionality I determined it was more beneficial to start over. Fortunately I was able to apply what I had learned working with the bad code and code the new application much faster. Once the application was coded for the second time went to test it and it had the same issue as before. Without allowing myself to get to frustrated I copied the files from the working test application that was printing headlines on screen and changed the file names to match my application and imported them into the project. Once I edited each of the files to match the new file names and extensions I tested the application. When the application loaded and I went to the Second View I noticed the scroll bar loaded on the side of the table but no headlines were loaded. I proceeded to scroll down the table and headlines started to show up, when I scrolled back to the top those headlines were printing as well. Currently this glitch is still present in the application, the headlines are printing fine but they will not display until the user scrolls through the table.

The total time spent this week on the project is:

- ◆ Looking over the code before starting a new application from scratch – 1.5 hours
- ◆ Writing the code for the new application – 2 hours
- ◆ Adding the files from the test application and editing them – 1 hour

Total time put into the project to date is 4 days 17 hours and 40 minutes

For the next week I am planning on coding in at least one if not two more websites for the application to print headlines for. I have a plan that only one SecondLevelViewController class will be needed for all three sites which would decrease the overall size of the application considerably. I hope this works but it will all depend on how well I know the language. I will be looking into why the application is not displaying the headlines until the user scrolls, I have no idea currently how I will be fixing this error but I will do my best to try.

Project Title: VG News!



Richie Thompson
November 3, 2009

This past week I have been working on coding the method to where all the websites will reference the SecondLevelView class. When I first set up a member variable to the class and used it locally to append to the URL within the function that accessed the URL. After learning the basics of how to append a string to another string and store them together as a single entity the URL opened and the headlines were still able to load correctly. Seeing that my idea would work I started to work on the next phase of this idea and wrote one line of code in the RootViewLevel implementation file to pass the site name into the string variable within SecondLevelView. This code generated a few errors as the variable was a protected class variable. Knowing how you would handle this problem in C++ I was not concerned that this issue would be difficult to overcome. I wrote a mutator function for the SecondLevelView class and called it in the RootLevelView implementation class. This idea would not work as the SDK was claiming there was no such function. I looked over the code several times to try and learn why the function was not being seen but I was unable to.

Without wanting to spend too much time on this issue I looked into a second option of declaring the variable public. This would prove to be much simpler, all I had to do was add the line “@public” above the variable and then the RootLevelView implementation file was able to change the value of the variable with no issues. This concept will add the ability for the application to have more user customization in that the user of the app can add their sites to retrieve headlines from. This is not part of the current project but if I continue to work with the application in the future this code will already be present and save me some time.

The total time spent this week on the project is:

- ◆ Writing the code to handle a new string for the site name and appending it to the URL – 1 hour
- ◆ Adding code to the RootLevelView to pass the site name to the SecondLevelView class and trying methods to fix the errors caused – 2 hours

Total time put into the project to date is 4 days 20 hours and 40 minutes

For this upcoming week I will be adding the two extra sites to grab headlines from to the application so three separate websites will be available in the application. I will be learning more about adding a second and third controller to the table on the root view so control can be passed to three different views from the main view. I will also be looking into adding a refresh button to the table view where the headlines are printed so the user can refresh the application to see if any new headlines are posted.

Project Title: VG News!



Richie Thompson
November 10, 2009

For the past week I have been working on adding in multiple websites for the application to load headlines from as well as adding a refresh button to the application. I realized rather early that I would need a loop to add multiple sites to the root view; this seemed to be the simplest method since all the sites would be the same in terms of design and navigation. I did some research for while loops in the objective c language and learned that it was very similar to while loops in c++ which made coding the loop very simple. The only problem I ran into while coding the loop was placement in the RootViewController.m file, I could get the loop to add the three instances of each site to the root view but they would all load the same headlines. After looking over the file I determined an alternate place to put the loop at and it worked like a charm. All three sites loaded and each site retrieved their specific headlines.

A final issue arose with the sites, Joystiq and QJ would each work fine with no issues and I could navigate to and from both but GamingBits would lock up when I went to scroll through the table so I omitted GamingBits from the list. After getting the code to work for the two websites and navigating through their respective headlines I began to look into adding a refresh button to the SecondLevelView file to refresh the headlines. This has proved to be rather challenging, I finally have been able to add the refresh button to the view but whenever the button is pressed the application crashes due to an uncaught exception. With getting sick this past weekend I was unable to fix this issue so the refresh button is still not working at this point in time.

The total time spent this week on the project is:

- ◆ Writing the loop to add two sites to the main view and determining the best place for it to implement at – 2 hours
- ◆ Adding the refresh button to the SecondLevelView.m file – 2 hours

Total time put into the project to date is 5 days and 40 minutes

For this week I will be working to fix the issues with the refresh button in the application so the application will no longer crash when it is pressed. In getting this portion of the application coded and working I feel like the application will be at a good final product and ready to go. It will not be perfect and there are still a ton of features I could add to the overall functionality of the application. With time running short to complete the project I will be unable to add any more features to the application. I still need to go back over my code and add comments to it so any one can look at my code and understand what it is I am trying to perform.

Project Title: VG News!



Richie Thompson
November 17, 2009

For the past week I have been working on adding in a refresh button to the application to refresh the headlines the application parses and update them on screen. I spent a good deal of time just researching into how to add a refresh button to the application. After researching was turning up very little to help me in my search I decided to look at the built in functions for the iPhone application that are precoded into the application. I was able to find exactly what I needed and with some additions to the code I had a refresh button implemented. Unfortunately all the refresh button would perform is crashing the application whenever it was pressed. I began to research this issue to see if a solution could be found but I had no luck as my code was correct but something, I could not determine what, was keeping it from working correctly.

Having worked on the refresh button for a while I decided my efforts were in vain so I scrapped the idea from the application. I would like to add this feature in later, but with time constraints I feel I would spend more time adding the button in rather than finishing the project. I started looking over my code to see if I could clean it up and make it a little more legible. In doing this I came up with an idea to move the code that printed the headlines on screen into another function. This move would prove very effective as the headlines are now immediately loaded on screen with no need to scroll to get them to display. I was very pleased to find a fix for this glitch, there is a slight delay before the headlines are printed while they are parsed but I feel the delay is manageable since everything prints immediately with no glitches.

The total time spent this week on the project is:

- ◆ Researching and coding a refresh button into the application – 1 hour 30 minutes
- ◆ Trying to solve the refresh issue before omitting it – 30 minutes
- ◆ Cleaning up the code and fixing the glitch of headlines not displaying – 30 minutes

Total time put into the project to date is 5 days 3 hours and 10 minutes

For the upcoming week I will spend the majority of time cleaning up the code and adding in comments. Presently there is very little commenting in my code which is not like me, I add as many comments as I can to my code. I feel like the project, though basic, is at a good stopping point. I would love to be able to add a lot more functionality and uniqueness to the design, but with time running short I feel like it is time to clean up the code and get the project ready to be displayed. I have truly enjoyed working on this project. I have learned so much about this new language and the process of designing an iPhone application. This is definitely a project I would love to pick up again and finalize my initial concept. I also hope to create a new icon for the application, while I do like the current icon that is in place I would prefer an icon that is a little flashier.

Project Title: VG News!



Richie Thompson
December 1, 2009

For the past two weeks I have been working on commenting my code and removing any unused functions that the SDK provided that was not necessary in the development of the project. My goal for commenting my code is to make the code as easy to understand as possible. I hope that anyone who looks over my code will be able to read the comments and know exactly what each function is doing when the application is running. The only part of the code that is not commented is the xml parses that I am using since I did not write this code I simply added in the site where I found the code, and the author of the code. I also implemented a new function in my application that allows the application to rotate the display if the phone is rotated by the user. I recorded a rough draft of the video for the software demo video for the application. The video does not have sound yet but that will be done soon enough.

The total time spent this week on the project is:

- ◆ Commenting the code – 2 hours
- ◆ Learning how to code the function to allow the allocation to rotate – 30 minutes
- ◆ Finding a screen capture software for macs and recording the demo video – 2 hours

Total time put into the project to date is 5 days 7 hours and 40 minutes

For what should be the last week of the project development I will be running the application several times to ensure that the application is functioning properly. I will strive to improve the icon of the application this week since I was unable to do so over Thanksgiving as I originally planned. I will be looking into some different options I could add to the application to help make it stand out a little more; this is in terms of visual advancements (i.e. different color scheme). I am not planning on spending too much time on adding in any new functionality. Finally I will be adding audio to the demo video and getting that finalized so it is ready to post on Youtube.

Project Title: VG News!



Richie Thompson
December 7, 2009

For the final week of the project I recorded the demo video for the software with a voice over so the video was easier to understand for the viewer. This proved to be quite challenging, I managed to find a nice quiet spot to record in but those little mistakes I made from time to time caused me to start over constantly. I also put together the posters for the presentation I will be conducting. For the posters I tried to include as much information as I could for all the components of the project, this way all the information is clear and anyone looking at the posters should easily see what I used to complete the project. I received help from Grace Cale in creating the poster design, she offered to help since she is good with designing, this came in handy since I am not always great at putting posters together. That is the majority of what I accomplished this week, I felt I should spend more time on the presentation to ensure I was completely prepared for it. I hope that anyone who is at my presentation will be able to get a good idea of everything my application is meant to accomplish, and when the time comes for it to be released for download I hope many iPhone owners download it and are able to get some use out of it.

The total time spent this week on the project is:

- ◆ Recording the Demo Video – 2 hours
- ◆ Drawing up rough sketches for the poster designs – 45 minutes
- ◆ Designing the posters and printing them off -

Total time put into the project to date is 5 days 7 hours and 40 minutes

My experience with the project has been a great one, granted at times I felt like I wanted pull my hair out. Learning a new language has really increased the overall challenge of this project for me, and though it may have resulted in a much smaller final product I feel like it was worth it just to get my feet wet with a new language. The biggest issue I faces was trying to solve errors when they popped up while I was coding. I could spend up to an hour looking over the code trying to determine where the error was at. I was pleased that in my final code my application has no errors in the code and it compiles and runs fine.

I have had little to no issues with this project from beginning to end. It really helped that I was able to select my own project at the beginning of the development process and stick to that concept. If the project was chosen for me I would not have had as much motivation to get it finished. My final product may have ended up being simple but it is my project and that gives me a sense of pride to know that. I hope that anyone else who takes on a project similar to mine will enjoy it and learn as much from it as I have. An iPhone application sure is a challenge for anyone to design. I have earned a lot more respect for the designers of the application I use on my iPhone.