 **CS 480**

Central Washington University

A Journalism iOS Application

 Authors: Gabriel McFann- Project Lead

 Kyle Hart- Documentation Lead

 Deep Bajwa- Lead Designer

 Date: November 11, 2013

 Client: Mr. Wilbert Ochoa

 Advisor: Dr. John Anvik

Table of Contents

Revision History…………………………………………………………………………………………………………………… 1

Introduction 2

**Project Vision 2**

**Real World Problem 2**

**Document Preview 2**

Project Management 3

**Process Model 4**

**Risk Management 4**

**Table 1: Risk Assessment Matrix 5**

**Work Breakdown Structure 5**

**Project Schedule 6**

**Software Development Tools 6**

Requirements 6

**Functional 6**

**Non-Functional 8**

**Feasibility Analysis 9**

Design 11

**Architectural 11**

Quality Assurance 11

**Document Standards 11**

**User Interface Guidelines 12**

**Change Control Process 13**

**Testing Process 13**

Conclusion 14

**Lessons Learned 14**

**Status Report 14**

Appendices 15

 Appendix A: Work Breakdown Structure 15

 **Appendix B: High Level Architectural diagram 16**

 **Appendix C: Sequence diagram 17**

 **Appendix D: Use Case diagram 18**

**Revision History**

|  |  |  |
| --- | --- | --- |
| **Revision Date** | **Changes Made** | **Name** |
| October 20, 2013 | Initial version of Iteration 1 | Deep Bajwa |
| October 20, 2013 | Initial version of Iteration 1 | Gabriel McFann |
| October 20, 2013 | Initial version of Iteration 1 | Kyle Hart |
| November 5, 2013 | Skype Meeting to revise requirements with Client | Deep & Gabe |
| November 7, 2013 | General Revision for 2nd Iteration  | Gabriel McFann |
| November 7,2013 | Compile Requirements Section | Deep Bajwa |
| November 11, 2013 | Architectural Design | Kyle Hart |
| November 11, 2013 | Revise Process Model | Kyle Hart |
| November 11, 2013 | Revise Conclusion | Kyle Hart |
| November 11, 2013 | UI Guidelines Section | Deep Bajwa |
| November 11, 2013 | Secondary Revision of Iteration 2 | Kyle Hart |
| November 11, 2013 | Insert Requirements | Deep Bajwa |
| November 11, 2013 | Revised document | Deep Bajwa |
| November 12, 2013 | Compile Document | Kyle Hart |
| November 26, 2013 | Adding correct page numbers | Deep & Gabe |
| November 27, 2013 | Risk Assessment Matrix Revision | Deep & Gabe |
| December 5, 2013 | Review Vision Statement | Gabriel McFann |
| December 7, 2013 | Outline Detailed Feasibility  | Deep Bajwa |
| December 8, 2013 | Insert Revised High Level Design | Deep Bajwa |
| December 8, 2013 | Revised Risk Management | Kyle Hart |
| December 9, 2013 | Use Case Table Created | Kyle Hart |
| December 9, 2013 | Use case diagram created | Deep Bajwa |
| December 9, 2013 | Sequence Diagram Created | Deep Bajwa |
| December 9, 2013 | Revised Document | Kyle Hart |
|  |  |  |
|  |  |  |
|  |  |  |

# **Introduction**

### Project Vision

The education of our nation’s children is directly influenced by our changing technology. Our team promotes the use of this new technology in the classroom.

### Real World Problem

With our Project Vision in mind, it came to our attention that Mr. Wilbert Ochoa had a need for an application for creating digital newspapers by his journalism students. He mentioned his need to our team because in his search for such an application he did not find a suitable solution. A pertinent requirement is the platform of iOS. Mr. Ochoa’s school district is testing the method of teaching with iPads and will soon deploy them district wide. The details are provided within the Project Management section of this document.

### Document Preview

The purpose of this document is to provide a written summary of the development process of our software solution. It will summarize the specification, design, testing, and documentation processes that we will implement over the course of development. Our intended audience includes the course instructor, Mr. Ochoa and maintenance developers. The major sections of this document to date are as follows:

* Project Overview – describes the factors affecting the project scope and requirements.
* Project Management –describes how we are going to create our solution.
* Requirements – describes the client’s need and essential task of the application.
* Quality Assurance – describes the standards for this project and document.
* Conclusion – includes a retrospective, project status report, and a list of priorities to focus on the next iteration.

Project Overview

Our goal is to create a native iOS application, optimized for iPad that will allow students to create, edit, and finalize a newspaper. The major group of users is middle school students ranging from 6th-8th grade. The client, as mentioned above, is Mr. Wilbert Ochoa. Mr. Ochoa is a teacher at Foothills Middle School located in Wenatchee, Washington. Mr. Ochoa is the major stakeholder. However, we as a team are also stakeholders. Mr. Ochoa is currently not aware of an existing iOS application, which can meet his specific requirements of his class. A custom application is more appropriate in this case. In our completed solution, students will be able to dynamically create their newspapers by sliding items around and instantly previewing how it looks. They will be able to edit the text directly. They will also be able to add photos. We will attempt to create a previewing system that will simulate a page flipping or a newspaper being turned.

# **Project Management**

Good software often requires a strong management that provides a good understating of development of the software, required skills, projected risks, timeline, scope, and quality of the project. Our team strives to have those skills. We believe that we can harness what we have learned in this class to create a plausible solution for Mr. Ochoa. Our project is divided into 3 roles- Project Lead, Design Lead, and Documentation Lead, which are described below.

|  |  |
| --- | --- |
| Team Organization | The project lead is responsible for making sure the team is completing tasks within the required time. The project lead will try to identify and solve problems to improve the productivity of the team.  |
| **Gabriel McFann – Project Lead** |  |

|  |  |
| --- | --- |
|  | The lead designer is in charge of the architectural and aesthetic part of our project. He oversees that design of the interface to the design of our class diagrams.  |
| **Deep Bajwa – Design Lead** |  |

|  |  |
| --- | --- |
|  | The documentation lead is responsible for documenting and submitting various processes and outcomes obtained in the development process. |
| Kyle Hart – Documentation Lead |  |

## Process Model

The Team decided to use an agile methodology, as it is more flexible to change. The application will be implemented in Objective-C using the Xcode IDE and Cocoa Touch Frameworks. An Agile process is best suited for this type of project given the potential changes from the client and learning curve for our team. Our team has had little to no exposure to Objective-C and so circumstances could change very frequently due to the nature of our project; namely, learning as we go. Along the same lines, we seek to get our software working early on and deliver what we can quickly rather than having a “perfect” solution. Another advantage to an Agile process is we are all students as well as developers. The team also has to adapt to our respective workloads as students to make the most of our time.

## Risk Management

It is important to document and prioritize the risks factors. In the development process, a programmer might experience different types of risks that can affect the productivity of the project. Some risks require higher consideration before implementing the next phase. Therefore, documenting and prioritizing the risks allows us to be aware of them and adapt to them accordingly.

During the planning phase, we acquired a good understanding of building the application. We weighed our different options, such as building a web application, versus a native. We concluded, given the accessibility, and the cost that we would be developing a native Objective-C application.

A major issue that wasn’t completely apparent to us is that none of our team members has any prior experience with Objective-C. This could cause major delays in completing the project on time. To minimize the impact, the team excessively is relying on tutorials available on YouTube as well as many other online academic resources. Our client is located in Wenatchee, which is roughly 70 miles away, which makes it hard hold meetings. To remedy this, we are using Skype, email, and phone. We are holding weekly meetings with our Advisor to minimize various risk factors and update the development of the project.

The client is expecting that each student in his class is going to have an iPad2 by the end of 2014. Since their iPads a little older, we could run into compatibility issues. We will be testing the application on at least two different iPads (iPad2 & iPad4), which will help us resolve most compatibility issues. Also the team did research and it appears that the compatibility risks are very rare. Similarly, if the client decides to change certain requirements at a point in development that may be near impossible to implement we would try to accommodate the client as much as possible, but we would also propose alternatives as to relieve undue stress upon us.

Since we will be testing the application on an iPad, there could be issues with getting complete access to an Apple developer account. In case of limited access or no access, it would not to be possible to effectively test and run the application. Therefore, to minimize the risk, the team has obtained the contact information of CWU’s mobile App developers to immediately seek their help to resolve any issues. Xcode does provide an emulator, which will let us test and run the application; however this is only so good.

If something terrible happens to the development machines, if they are stolen, or if something goes wrong with the code, we could lose a lot of headway. To overcome with this type of risk, we frequently save our code on our repository. If one of us got sick, it could slow the development process. By way of understanding those risks, a team member who is unavailable would try to work from home if possible. Since we work collaboratively, we try to understand everyone’s part of the work. If in case a certain team member becomes unavailable the rest of the team will still share that knowledge and will try their best to work effectively.

Below is a risk assessment table that would be used to document and prioritize risk factors as we implement the program.

## Table 1: Risk Assessment Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Descriptor  | Minor | Major | Extreme |
| Client’s Location versus our own | √ |  |  |
| Access to CWU Apple Developer account Denied | √ |  |  |
| Learning an New Language |  | √ |  |
| A Team Member becomes ill and cannot contribute |  | √ |  |
| Program not compatible with older iPads (2) |  | √ |  |
| Theft of development platform (Macs) |  |  | √ |
| Failure of Development Platform (Macs) |  |  | √ |
| Requirements change so much that we cannot meet our goals |  |  | √ |

## Work Breakdown Structure

The breakdown of our project is partially taken from our team roles however, in the words of the Three Musketeers, “All for one, and one for all!” The following is a list of current milestones and tasks. We have gained access to the license of CWU’s Apple developer account. Since we are in the early development stage, we are utilizing our time learning Objective-C. We have gone through many Objective-C tutorials on YouTube and other academic sources that are available online to get our feet wet and we will be coding soon.

## Project Schedule

By the end of the next quarter, we will have a working application. The goal is to build an iOS based prototype that would handle a user login and establish an administrator account, which can add or delete users. We would create test cases and use code coverage tools to meet the coverage requirements.

Starting winter quarter in CS481, we hope to have a working application that will have all the essentials features described in the requirement section. For instance, students must be able to edit and format text directly in the application. If we have enough time available, we would implement the integration of a Google Drive into the application. We will try to include as many desirable and optional features as we can. For instance, spell checker and a notification feature. The desirable features are described in a requirement section.

## Software Development Tools

We are going to use various software development tools that would help us build an effective and efficient application. Below is a list of software development tools.

* Issue tracking system – Jira
* Version control system – Mecurial (Atlassian Source Tree)
* Integrated development environment – Xcode
* Code coverage tool – Currently looking into GCov and Cover Story
* Designing Tool – Microsoft Visio, Pencil, and GIMP
* Documentation tool – Microsoft Office such as Word, Excel, Publisher, and Project

# Requirements

The project is designed to be an iOS based application, which requires a Macintosh and the Xcode IDE to develop the application. The requirements are provided by the client, which is split up into two main categories – functional and non-functional. Furthermore, the functional requirements are divided into two types – essential and desirable tasks. The team first decided to implement the core requirements specified by the client. The team will try to implement desirable features depending on the available time.

## **Functional**

The main functional requirement of the application is that the students should be able to create, edit and submit their work within the application. The application will provide a common platform between students and their teacher. The students will be able to view other students’ work and the teacher will be able to correct and grade their submitted work.

The students will be able to export their work to a PDF format. They will have the ability to view the work submitted by their classmates and students in the other classes. They will be able to add photos and create hyperlinks. As they will be writing small paragraphs, the application will have a built in dictionary that can auto correct misspelled words and look up meaning of specific words. Notification of upcoming assignments and deadlines will be sent to the student as to keep him/her on track.

As for now, we plan to use MySQL as our database tool to account for students’ information and other pieces of data that need to be saved. In our opinion, having students’ work saved on a database, would restrict students to make changes after submission or make any unnecessary changes to other students’ work. The data will be stored in a MySQL Workbench 5.2 CE database. Alternatively, the application will be secured so that only authorized users can use the application.

As we implement the program, we will make sure that the design should be easy modify for future use and leave appropriate comments in the program so that it becomes easier for new developers. Furthermore, the functional requirements are broken into two categories – essential and desirable tasks. The main objective of the project is to deliver the essential tasks requested by the client.

Table 2: Time Schedule of Essential Tasks

|  |  |
| --- | --- |
| **Task** | **Estimate time of completion in weeks** |
| Edit newspaper | 2 weeks – January 19, 2014 |
| Format the newspaper in different ways | 2 weeks – February 2, 2014  |
| Edit the text directly | 2 weeks – February 16, 2014  |
| Add photos | 2 weeks – March 2, 2014  |

## The detailed functional requirements are shown in the table below, which provides a sufficient level of detail for a designer to design a system satisfying the requirements. Also, for a tester to verify that the system satisfies requirements.

Table 3: Detailed Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Description** | **Priority** | **Estimated Effort** |
| 1.0 | The student is able to log into and out of the system | High |  |
| 1.1 | The student able to access the student page | High |  |
| 1.1.1 | The student is able to create new documents | High |  |
| 1.1.2 | The student is able to access saved documents | High |  |
| 1.2 | The student is able to edit their document | High |  |
| 1.2.1 | The student is able to edit specific text | High |  |
| 1.2.2 | The student is able to add/ remove pictures | High |  |
| 1.2.3 | The student is able to format the paper in different ways | High |  |
| 2.0 | The admin is able to log into and out of the system | High |  |
| 2.1 | The admin can see all the classes that are registered with the app | High |  |
| 2.1.1 | The admin can see every student within each class | High |  |
| 2.1.2 | The admin is able to read all of the papers by most recently submitted | High |  |
| 2.2 | The admin can add/ remove classes | High |  |
| 2.2.1 | The admin can add/remove students from each class | High |  |
| 3.0 | The student is able to look up and define a word using built in dictionary | Moderate |  |
| 3.1 | The user is able to run a spell check before submitting their newspaper | Moderate |  |
| 4.0 | The student is able import media files to the newspaper | Low |  |
| 4.1 | The student is able to play the imported files within the newspaper | Low |  |
| 5.0 | The Student is able to see deadline notifications for when their newspaper is due | Low |  |

## **Non-Functional**

The main non-functional requirement is that our application will be optimized for an iPad. Since we are building an application for middle school students, our application will be flexible to use for students and a teacher for their intended use. Since the project is going to offer a touch user interface, it’s not going to have a tooltip for various buttons and functionalities. Also, considering the level of understanding and experience of students, the client recommends the use of icons. Below is a list of the basic non-functional tasks. The students will be able to export their work to a PDF format.

* The application will be designed with the students skill level in mind, it will be made to be as simple and intuitive as possible
* Use of icons to help visualize the application’s features

**Design constraint**

* iOS based application optimized for iPads

## **Feasibility Analysis**

By having a good understanding of feasibility, we will be able to identify the important risks associated with the project that must be addressed. The risk increases dramatically if we are less familiar with the programing language and the system. In order to deliver an effective and efficient application, the team is focusing on learning Objective-C and delivering the essential tasks upon completion. Therefore, based on the key requirements, timeline, and considering the team’s knowledge, the essential tasks will be given higher priority than the desirable tasks. As mentioned earlier, since we broke down each task in a two-week period, we will be able to effectively implement the tasks and manage the project development. This would allow us to grasp a good understanding of the program as it gets more complex.

We will consider the compatibility of our application with the older version of the iPad and integrate the application with the MySQL database. This would provide us the opportunity to check if the data successfully flows within the application and the database. Throughout the development of the project, we would consult with our advisor if certain tasks or technical approach is feasible from a technical perspective.

The main goal is to complete the essential tasks for the client by the end of winter quarter (2014). Below is a project prototype (*Figure 1*) that will handle user logon and allow the students to create newspaper using the available functionalities provided in the application. On the other hand, the client will have the ability to add or remove users if they happened to drop out of the class. The client would be able to edit and grade students’ work as well.

**iNewsies**

Figure - User Interface

#

#

Class 4

2 AM

Class 3

12 PM

Class 2

9 AM

Class 1

8 AM

Section 1

Section 2

See Below Figure 2

**Published Newspaper**

Kayla

Sunny

James

Amanda

Bobby

After the students successfully login, they will be able to choose their class as shown in Section 1 of the figure. Section 2 displays the most recently submitted newspapers from all different classes, which provides the students with multiple newspapers to read over should they so desire.

Beside that as the students choose their class, they would be able to create their own newspaper and have the ability to view other students’ work in that class as shown in *Figure 2*. *Figure 3* shows the basic functionality of the application.

**Class 4**

**Create Newspaper**

**Published Newspaper of Class 4**

Robin

Jim

Jon

Karle

#

# http://c.dryicons.com/images/icon_sets/colorful_stickers_part_5_icons_set/png/256x256/news.png

#

Sheet 3

Sheet 2

Sheet 1



Read/Publish

Home

Save

Import/Export

Background

New

Figure 3 - Application Functionalities

# Design

#

## **Architectural**

For our architectural design, we are using a Model-View-Controller system. An example of the system: we have one model named Articles for each article that is written for the newspaper. That model would be encapsulated within another model called Article Collection, which would hold lists of all the articles that are to be displayed on each newspaper. On the other hand, there will be two other models – Image Collection and View Collection that would contain images and views accordingly. The View Collection will call the models and use them to draw the newspaper out, on our UI for the user to see. The last model will be a Newspaper model, which we will use to create the newspaper and it would encapsulate Article Collection, Image Collection, and View Collection models. In the diagram in Appendix C, we show a general view of a high level design and how it is going to work together. This Design Pattern allows for good extendibility.

# Quality Assurance

To develop a high-quality solution for a client, we will prevent mistakes and errors to lower the risks. Alternately, the team has researched the requirements and guidelines in order to successfully publish the application on Apple’s App Store. Apple has strict guidelines that must be followed to protect their products’ identity and maintain the same application standards. For instance Apple restricts the use of their logo, product names, and the shape of the icons or labels must have rounded corners. Besides that, the team works collaboratively to produce an efficient and an effective application. The team held weekly meetings with their client and advisor to get feedback and update the project development. Furthermore, to test the project demonstration, the team used an iPad simulator integrated in Xcode to test the expected outcome.

## **Document Standards**

This Project requires standards in order for us to be successful in the creation of our software solution. To date our quality assurance includes standards for our document.

When editing our document we will require at least one revision. All team members will always review our document at least 48 hours before the submission deadline. After every team member has given feedback on the sections that they were not assigned to write, a final revision is made, then finally the document lead will submit it.

## **User Interface Guidelines**

When working with a gesture based device, developing for this environment can be tricky yet natural. Tricky in the sense of the question: “What is natural?” Our interface will not be too innovative in the use of these gesture-based systems; however, it will take advantage of standards that have been established. We will make the interface as easy to use as possible. We read some of the important guidelines and protocols that are necessary for developing an iOS application. We learned that an application would need to follow guidelines for Cocoa, which is the framework that contains a dynamic shared library along with the associated resources, such as xib (or nib) files, image files, and header files. By default, the application is linked to the Foundation, UIKit, and Core Graphics framework capabilities, which are accessed through header files. Since the library is dynamically shared, multiple applications or tasks can access the framework code and resources simultaneously.

On the other hand, in terms of design requirements, an application must adopt to iOS Human Interface Guidelines. For instance, iOS 7 embodies the three main themes – deference, clarity, and depth. Deference states that an application must let the content extend to the edges of the screen as similar to the Weather and Maps application. In order to have a good clarity, the functionality must be clear and easy to interact with, which encourage developers to use plenty of negative space as similar to Message application. Alternatively, the content should appear in distinct layers that convey hierarchy and position as similar to Calendar application.

Apart from that we studied some of the editing applications available on App Store such as Paper, which was recommended by our client. We studied the various interface designs. We learned that iOS applications tend to be graphically rich. Therefore, an application must have precise and lucid icons. The icons must have rounded corners and sharpened focus to motivate the design. The photos and graphics are required to use their original aspect ratio, and the scale should not be greater than 100%. For iPad 2 the standard resolution for App icon is 76 x 76 pixels. Also, the text should be legible at every size. Beside that images that replicate Apple products are prohibited under copyright laws. On the other hand, the students that will be using our application will need basic computer skills to navigate through it. Since the students are in a learning environment this should not be too difficult for them to learn and be familiar with the application functionalities.

## **Change Control Process**

Since we are getting close to the end of the quarter and into heavy development, we may encounter changes that may be unnecessary and/or detrimental to our project. In order to combat these digressions we will discuss the essential requirements at every meeting. This will help solidify in our minds our main goal. When something that may be contrary to that goal arises, we will be able to stay focused and not be distracted by it. We are using agile process since the requirements tend to change over the course of the project. We defined the specifications in the form of user stories that were agreed upon by the team members. In the weekly iterative review meetings, team members planned how to complete these stories and how many iterations to assign for each story. This will allow us to easily make changes during implementation instead of redesigning the project. Since we work to track the progress for each iterative cycle. Alternatively, we are using Jira as our issue tracking system, which enabled us to capture issues, plan work, and resolve issues. We create new issue(s) if a team member is experiencing a problem with a certain task or trying to resolve a bug. The other team members can view the existing issues, which they try to resolve it to keep the project development steady.

## **Testing Process**

We will perform our initial system level testing using an iOS device such as an iPad and built in emulator integrated into Xcode. This particular device (iPad) had to be registered with the CWU Apple developer account. We created a paper prototype, which we presented to the client. We received a positive feedback from the client and some suggestions to change the layout of the application. For instance, the client wants to use various tiles for each student and class. We planned to use four different testing techniques – White Box, Integration testing, Acceptance testing and Installation testing. In order to meet the client’s requirements, we will use acceptance testing to test high-level tests created by the client to test the completeness of a solution for a user story. It is important for us to test various specifications before we publish our application on Apple’s App Store. By having all the tests pass will help us in determining the team’s progress in the right direction. Alternatively, in order to detect any potential failure scenarios, we plan to use White Box testing. This will help us determine the suitable input data for testing various APIs and specific code paths that need to be tested by analyzing the source code to verify that the application block is able to meet all functional requirements.

On the other hand, we planned to use integration testing to ensure that the application meets the essential requirements. Since we are going to use a database to store student’s data, it is important to test if the database actually works with our application. This will help us determine if there is any incompatible or incorrect data types, errors, failure to trap exceptions, and mismatch of data types that will minimize the risk of losing data. Alternatively, we are going to use installation testing on iPad – 2 and other newer version of iPads to check any compatibility. As we get closer to finishing our project, we will choose a few students from our client’s (Mr. Ochoa) class to test the application on their iPads. We will make any necessary changes considering the feedback from students and our client. Besides that we will be using the XCTest framework built in Xcode to create the test cases. We will develop various test suites to test different aspects of our code.

# Conclusion

The problem we are trying to solve is that there is not a good solution to help students create, edit, and view newspapers on an iOS device, i.e. an iPad. We plan to solve this problem by developing an app that will meet Mr. Ochoa’s needs.

## **Lessons Learned**

 In retrospect, with the academic load that are team currently undertakes, it has been hard to nail down an official meeting time. However, we do meet on the fly, which is better than no meeting. On a high note, we as a team get along very well and work democratically. The team had experienced a lack of inter-team communication, which we will like to improve in future. On the other hand our communication with our client has greatly improved in the fact that we used to have a Skype meeting every 2-3 weeks, now we are having a weekly Skype meeting with Mr. Ochoa.

## **Status Report**

In this iteration, we held weekly Skype meetings and exchanged e-mails with our client to confirm the requirements and to report on our progress. Alternatively, we met him personally and showed him the progress of the project. We received positive feedback from client.

# Appendices

# Appendix A: Work Breakdown Structure



## Appendix B: High Level Architectural diagram

Newspaper

Article Collection

Articles

Images

Views

View Collection

Image Collection

## Appendix D: Use Case diagram

