Software Plan

Client: Dr Darren Lim, Assistant Professor

Proposed Project: *Java Online Learning Toolkit (J.O.L.T.)*

Delivered by: *518 Interactive*

Team Members:
Lawrence Gregory
Christopher Hughto
Erik Stegmann
Connor Vander Bogart
Jedidiah Turnbull

Revision: 1.1
Revised: 10/20/09
Date: 09/24/09

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to formally define Dr. Lim's problem, and offer a generalized solution. This document serves to ensure that both *518 Interactive* and Dr. Lim understand and agree to the problem and generalized solution.

1.2 Scope

This document outlines *518 Interactive's* current understanding of Dr. Lim's problem.

1.3 Audience

This document is intended for the client, Dr. Darren Lim, as well as Dr. Eric Breimer, Dr. Timoth Lederman, students of the Fall '09 Software Engineering class, and members of the *518 Interactive* Development Team.

2 System Definition

2.1 Problem Definition

The client, Dr. Lim, is very busy. Dr. Lim has a limited amount of time to accomplish his daily tasks. One of Dr. Lim's most time consuming responsibilities is the grading of his students' programming assignments. Currently, Dr. Lim is forced to accept paper and digital copies of each student's program. Dr. Lim then executes each attempted solution by manually running them one at a time and comparing the output. Not only is this an inefficient method of grading, it is a waste of paper.

As a result of this time consuming process, students do not receive the prompt feedback that they often wish to have. The entire grading process is impaired, which hurts both the students and the faculty.

2.2 System Justification

The System, titled *Java Online Learning Toolkit (JOLT),* will be of great assistance to the client. JOLT will allow for the entire programming project process to be streamlined and automated. Students will receive instant feedback about their solutions. Faculty will also have an automatically generated, up-to-date grade book for all students that they instruct.

2.3 Goals for System and Project

The goal for JOLT will be to have a fully-functional, web-based environment which will be accessible from any computer with an Internet connection and support for any of the mainstream browsers listed in section 2.4.

JOLT shall implement a flexible framework that will allow students to view a variety problems created by their professor. Students shall have the ability to solve the problems and receive instant feedback about their solution attempts.

2.4 Constraints on System and Project

Due to the variety of configurations in the personal computers of the faculty and students, JOLT must be cross-browser compatible. At the very least, JOLT should work with the latest version of the following: Microsoft Internet Explorer, Mozilla Firefox, Apple Safari, Opera, and Google Chrome. This implies that no browser-specific features, such as Active-X Controls, can be used for any part of JOLT.

JOLT will only accept code written by students in the Java™ programming language, per request of our client.

2.5 Functions To Be Provided:

JOLT shall provide the following functions via a web interface. A complete function list shall be provided in the *Software Requirements Specification* document.

JOLT will utilize a database, most likely MySQL, to store and retrieve data. A PHP (PHP Hypertext Preprocessor) and HTML (HyperText Markup Language)/CSS (Cascading Style Sheet) based website will comprise the user interface for the System.

The Java SDK (Version 1.6), and runtime environment (Version 1.6) shall be provided to run the code submitted through JOLT.

2.6 User Characteristics

Two classes of users have been identified for JOLT: Faculty and Students.

Faculty will have the ability to:

- Create problems and test data for each problem.
- Search/Assign problems created by themselves and other faculty members.
- Have an online "Grade Book" for viewing student's progress.
- Generate reports. No specifics are known at this time.

Students will have the ability to:

- View problems assigned to them.
- Submit Java™ source code to solve assigned problems.
- Receive immediate feedback on their solution attempts (pass/fail)
- Have an online “Grade Book” for viewing their progress.

2.7 Environments

We have defined the following environments for JOLT:

2.7.1 Development Environment

The development environment is comprised of all the hardware and software which *518 Interactive* will use to develop JOLT. The details of the environment are as follows:

Operating System: CentOS (Linux) Release 5.2 (Final)
Server Name: oraserv.cs.siena.edu
CPU Type: x86_64
Web Server: Apache Version 2.2.9
PHP Version: 5.2.6
Database: MySQL Version 5.0.45; Oracle Version 9.0.1

Windows Machine:

- Operating System: Microsoft Windows Vista Enterprise
  - Build: 6002
  - Revision: 18005
  - Service Pack 2
- Processor: Intel® Core™2 Duo CPU
  - Model: E7500
  - Speed 2.93 GHz
- Memory (RAM): 4.00 GB
- System Type: 32-bit
- Dual Monitor Setup
- Software Installed:
  - Microsoft Office 2007 (Including Microsoft Project)
  - Macromedia Dreamweaver, Fireworks, Flash , Freehand, Studio (2004 Versions)
  - Internet Explorer, Mozilla Firefox, Google Chrome

Macintosh Machine:

- Operating System: Apple Mac OS X
  - Version 10.4.11
- Model: iMac5
- Processor: Intel Core2 Duo
  - Speed: 2 GHz
- Memory (RAM): 1.00 GB
- Dual Monitor Setup
- Software Installed:
  - Microsoft Office 2004 for Mac
  - Macromedia Dreamweaver, Fireworks, Flash, Freehand, Studio (2004 Versions)
  - Safari, Mozilla Firefox

2.7.2 Operating Environment

The operating environment is comprised of all the hardware and software which the Production version of JOLT will be run on. The specifics of the operating environment are not determined at this time. However, the environment must be able to handle the constraints defined in section 2.4

2.7.3 Maintenance Environment

The Maintenance Environment is comprised of all the hardware and software which will be used to make modifications to the software after it has gone into Production. The specifics of the maintenance environment are not determined at this time.

2.8 Solution Strategy

To effectively develop a solution for Dr. Lim, *518 Interactive* has decided to use a modified version of the *Classic* Waterfall Model, which is a type of the Linear Sequential Model.

Software Plan: *518 Interactive* will define Dr. Lim's problem after a series of meetings with the client to develop the overall picture. A formal document is created which outlines the problem and offers a tentative, generalized solution. Deliverable: *Software Plan*

Analysis: Through further interaction with Dr. Lim, *518 Interactive* will detail his specific needs. Based on those needs, *518 Interactive* will define the functional and non-functional requirements. Deliverable: *Software Requirements Specification*

Design: A prototype of the System will be developed for Dr. Lim by *518 Interactive* which will design a strategy for implementing JOLT using the *Software Requirements Specification.*
Deliverable: *Preliminary Software Design*

Detailed Design: *518 Interactive* will revise the preliminary software design and meet the established functional and non-functional requirements. Deliverable: *Detailed Software Design, Test Plan*

Acceptance Test: JOLT is run through a series of tests which will show whether or not the functional and non-functional requirements have been met. Deliverable: *All Acceptance Test Results.*

The above outlined process shall be repeated when changes to JOLT are required or requested.

2.9 Priority of Features

The ability for Students and Faculty to log in to their accounts is crucial. Faculty must have the ability to create and assign problems, and the students must have the functionality to submit their solutions and receive feedback.

The least important features relate to the style and design of JOLT, such as graphics, and AJAX (Asynchronous JavaScript and XML) implementation.

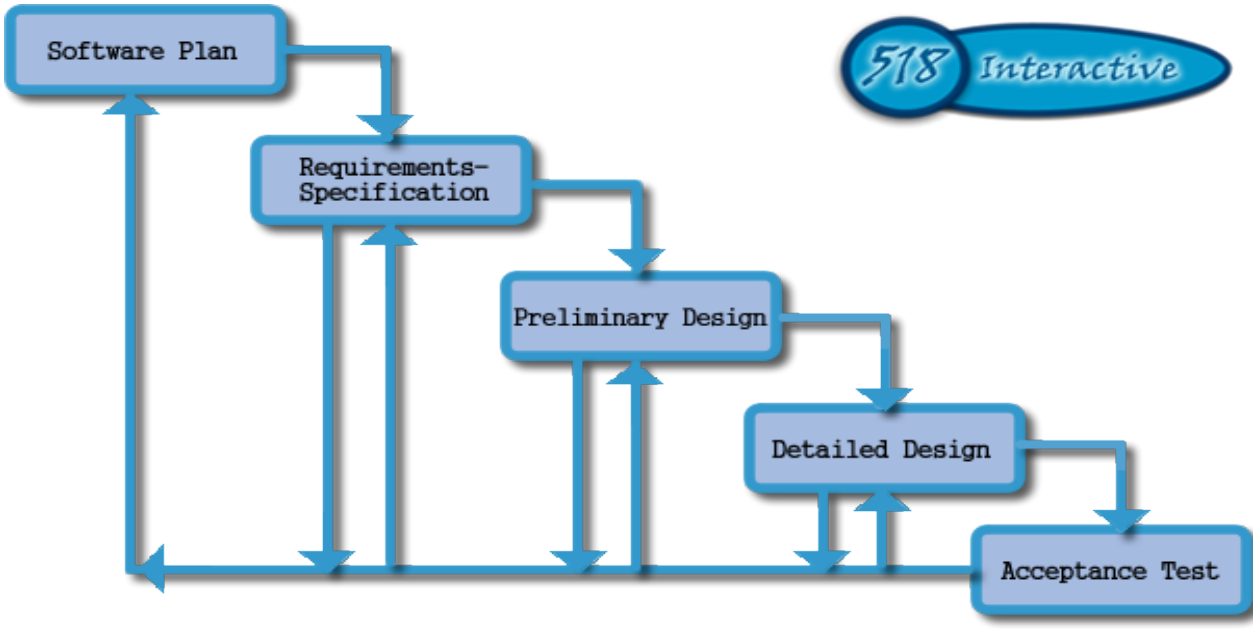
2.10 System Acceptance Criteria

In order for JOLT to be accepted, it must successfully perform all functional / nonfunctional requirements as specified in the Software Requirements Specification. A formal test document, called the "Acceptance Test" shall be delivered to Dr. Lim outlining all results of the test.

3 Project Plan

3.1 Waterfall Model

This is a sequential method of organizing tasks in a software engineering project.



This is our version of the Waterfall Model. Each stage has the ability to backtrack to any previous stage, or can proceed to the next stage when complete. Note that it is not possible to "short-circuit" the model by jumping over any step.

Please refer to Section 2.8 (Solution Strategy) for a description of each step.

3.2 Organizational Structure

The *518 Interactive* Team:

Name:	**Email:**	**Phone Number:**
Gregory, Lawrence	le30greg@siena.edu	(518) 567-9345
Stegmann, Erik	ej15steg@siena.edu	(518 758-2189
Turnbull, Jedidiah	jm29turn@siena.edu	(518) 782-6084
Hughto, Christopher	cj12hugh@siena.edu	(518) 235-8550
Vander Bogart, Connor	cp21vand@siena.edu	(518) 283-1992

Team Positions:

Gregory, Lawrence **Team Leader:**

It is the Team Leader's duty to organize and oversee team and client meetings. The Team Leader is also responsible for overseeing and managing resources to ensure that the product is finished by the required deadlines and in the desired manner ensuring that the client's needs are met.

Stegmann, Erik **Lead Graphics and Interfaces Designer:**

The Lead Graphics Designer develops the graphics for the websites and JOLT. He also ensures that the user and client alike are greeted with an intuitive and creative layout that is easy to navigate and clear to even the most inexperienced of users.

Turnbull, Jedidiah **Systems Administrator:**

The Systems Administrator manages the hardware resources for the *518 Interactive* Development Team. These resources include but are not limited to: PCs, software, profiles, and systems. It is the duty of the Systems Administrator to ensure the computers are reliable and readily available for use. The Systems Administrator also performs system upgrades, software updates and program installations as necessary.

Hughto, Christopher **Webmaster:**

The Webmaster manages the company's website, creates layouts and designs that reflect *518 Interactive's* professional image to its clients. The Webmaster ensures that the website is up to date so that the *518 Interactive* development team can use it as a hub for information

Vander Bogart, Connor **Organizational Information Manager:**

The Organizational Information Manager's duty is to take notes and minutes of meetings. The Organizational Information Manager ensures all documentation is up to date so that *518 Interactive* has correct, pertinent information when needed. The Organizational Information Manager's documentation includes the Software Plan, Software Requirements Specification, Preliminary Design, Detailed Design, and the Acceptance Test.

3.3 Staffing and Resource Requirements

To complete the development process for JOLT, *518 Interactive* will require a team of five (5) personnel (referenced in Section 3.2). The team will be responsible for both the graphical interface as well as the backend that drives JOLT. The team will require approximately 12 weeks in order to complete the Software Requirements Specification and Prototype. Another 12 weeks will be required to develop the Production version, which will have to pass an Acceptance Test.

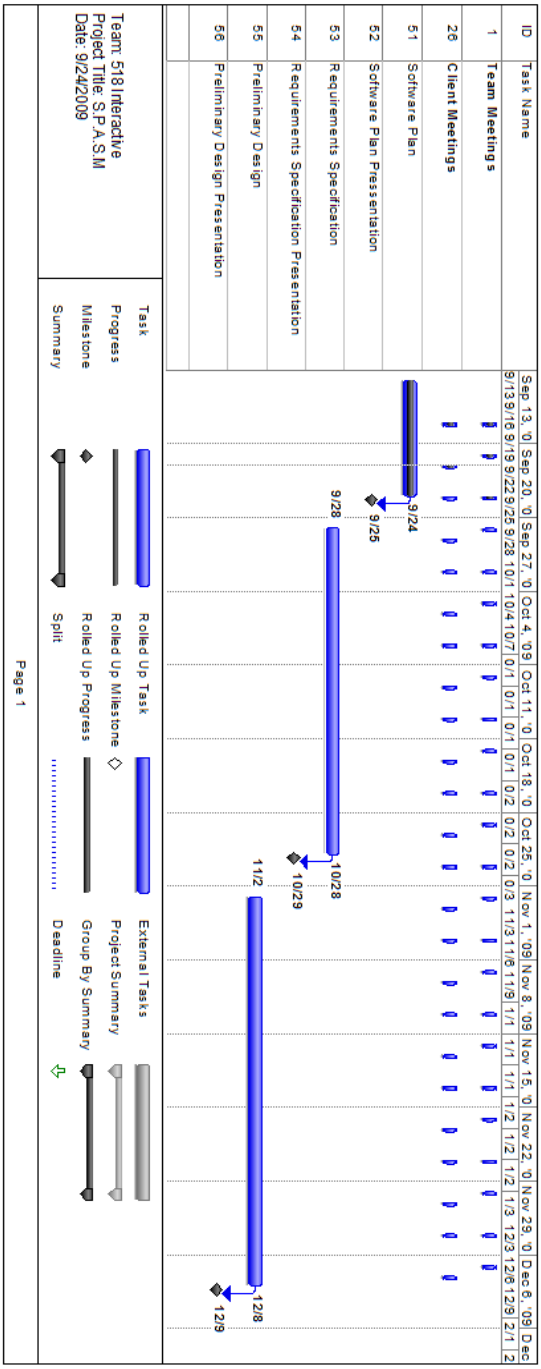
The staff of *518 Interactive* is diverse in ability, which should eliminate the need to contract work out to other vendors. However, external resources are readily available if the need arises.

An effective implementation of JOLT will require a server running any operating system that is capable of handling an Apache web server, MySQL database, and Java SDK & runtime environments.

To begin development, *518 Interactive* will be using the computers in the Software Engineering Lab in Siena College's Computer Science department (Roger Bacon, 3rd Floor). The *518 Interactive* Development Team will be using both Microsoft Windows and Apple Macintosh based computers to design and implement JOLT. The details of the systems are listed in Section 2.7.1.

3.4 Development Schedule

The tentative development schedule is outlined by the Timeline (Gannt Chart) on the following page. Changes are made as needed by agreement between *518 Interactive* and Dr. Lim.



3.5 Project Monitoring and Control Mechanisms

Throughout the development process, *518 Interactive* will meet with Dr. Lim on a bi-weekly basis. This will ensure that the needs of Dr. Lim are put first. The meetings will also ensure that *518 Interactive* has a correct and complete understanding of JOLT's requirements.

Dr. Lim will also have the opportunity to analyze and critique the Software Plan, Software Requirements Specifications, Preliminary Design, and Detailed Design to ensure that JOLT is implemented to his specifications.

3.6 Tools and Techniques to be Used

To effectively manage time and resources, *518 Interactive* will be using Microsoft® Project to timeline projects and deadlines. In addition, the Adobe® Creative Suite (including Dreamweaver, Flash, and Fireworks) will be used to develop JOLT. Other tools and techniques may be used by *518 Interactive* as deemed appropriate.

3.7 Programming Languages

518 Interactive may use PHP, MySQL, HTML, and CSS to develop JOLT. Java will be required as part of JOLT, but will not be used by *518 Interactive* to create JOLT. Other languages may be used by *518 Interactive* if deemed appropriate.

3.8 Testing Requirements

Testing of JOLT shall be done by *518 Interactive* throughout the development phase. A final acceptance test will be run, which will ensure that all features outlined in the Software Requirements Specification work correctly.

3.9 Supporting Documents Required

Supporting documentation for JOLT will be provided to Dr. Lim at each stage of the development process, to keep him informed of the progress of the project and the development of the software.

Supporting Documentation includes:
Software Plan - September 24, 2009
Requirement Specifications - October 28, 2009
Preliminary Software Design – December 8, 2009

Other required documents include the Detail Software Design and Acceptance Test. The additional documents will be delivered in the Spring 2010 semester. Delivery dates have not been finalized at this time.

3.10 Demonstration and Delivery

518 Interactive will deliver to the client the supporting documents outlined in section 3.9. Along with each document will be a presentation by *518 Interactive*. The final Acceptance Test, once completed satisfactorily, will mark the delivery of JOLT.

3.11 Method and Time of Delivery

Once JOLT passes the acceptance test, all source code and other required files shall be transferred and installed to Dr. Lim's production environment.

3.12 Sources of Information

518 Interactive will be obtaining information from Dr. Lim over the course of the bi-weekly client meetings. This information will allow us to better meet Dr. Lim's needs and solve his problem in the desired manner. Additional sources of information will be Dr. Lederman and Dr. Breimer regarding software and website development respectively. Prior Software Engineering projects will also be researched to get a general outline of the way things should be accomplished.

4 Appendices

4.1 Team Resumes

4.1.1 Lawrence Gregory (Team Leader)

Lawrence Gregory
72 Stottville Road
Hudson, NY 12534
518.567.9345
larry@legit-solutions.com

OBJECTIVE

To obtain a challenging position in the field of computer science.

EDUCATION

Siena College, Loudonville, NY
B.S. Computer Science, May 2010
GPA: 3.54; **Computer Science:** 3.56; Upsilon Pi Epsilon Computer Science Honor Society

COMPUTER EXPERIENCE

Languages: C++, Visual Basic, x86 Assembly, Java, PHP, XHTML, COBOL
Operating Systems: UNIX, Z/OS, Windows 9x/NT/2000/XP/Vista/7
Software Packages: Microsoft Office 98 – 07, Microsoft Visual Basic, Microsoft Visual C++, Eclipse, Macromedia Dreamweaver, QMF for Windows
Database Management: Oracle 9i, DB2, Microsoft SQL Server, MySQL

RELEVANT EXPERIENCE

Team Leader, **Software Engineering I**, Loudonville, NY, September 2009 – Present

- Organized and supervised group during software planning process
- Delegated workload effectively among team members
- Established group meetings and project reviews with client and class instructor
- Helped design team website

Applications Programmer Intern, **NYS Division of the Lottery**, Schenectady, NY, May 2008 – Present

- Designed, built, and maintained COBOL programs with/without database involvement
- Collaborated with fellow programmers to effectively solve problems
- Wrote, executed, and documented test plans for programs.

ADDITIONAL EXPERIENCE

Customer Service Manager (CSM), **Wal-Mart Stores, Inc.**, Hudson, NY, 2006 - Present

- Supervised front-end personnel to ensure company policies were being followed
- Administered probationary and annual evaluations to all front-end personnel
- Delegated tasks as needed
- Executed register audits to ensure proper cash control
- Diffused difficult situations with unruly patrons

4.1.2 Connor Vander Bogart (Organizational Information Manager)

Connor VanderBogart
38 Willow Lane
Poestenkill, NY, 12140
518-229-5745
cp21vand@siena.edu

Objective

To achieve a position in the field of Computer Science that will implement my knowledge of programming and/or computer troubleshooting/repair.

Education

Siena College, Loudonville, N.Y.
B.S. in Computer Science, Minor in Business, May 2010

Relevant Courses

Intro to Computer Science, Intro to Programming, Data Structures, Assembly Language and Computer Architecture, Object-Oriented Design and Programming, Analysis of Algorithms, Data Base Management, Web Design, Calculus I, Calculus II, Discrete Structures I, Discrete Structures II

Professional Experience

***Organization Information Manager,* Software Engineering I**, Loudonville, N.Y., September 2009-Present

- Taking notes/minutes at team meetings and client meetings to help inform team members
- Organizing documents related to the team project/Keep info up to date

New York State Police Internship, NYS Police Headquarters, Albany, N.Y., September 2009-Present

- Collaborating with superiors to achieve goals
- Devising solutions to web-based problems/Assisting IS members with quandries
- Improving upon services used by NYS Police and outside users alike

Employment

***Front End Supervisor,* Price Chopper**, Brunswick, N.Y., 2004-2008
***Front End Shift leader,* Hannaford**, Wynantskill, N.Y., 2008-Present

- Utilizing leadership skills to achieve goals for a large group of people
- Improved status from cashier to management position through hard work

Computer Skills

Programs: Microsoft Word, PowerPoint, Excel, BlueJ, Dreamweaver, Fireworks
Languages: Java, C, HTML, XML, PHP, JavaScript, AJAX, Linux, Visual Basic

4.1.3 Jedidiah Turnbull (Systems Administrator)

Jedidiah Turnbull
479 Ridge Rd.
Campbell Hall, NY 10916
845.705.7199
jm29turn@siena.edu

OBJECTIVE

To obtain a challenging position in the field of computer science.

EDUCATION

Siena College, Loudonville, NY
B.S. Mathematics and Computer Science, May 2010
GPA: 3.65; **Mathematics:** 3.53; **Computer Science:** 3.80
Upsilon Pi Epsilon Computer Science Honor Society
Pi Mu Epsilon Mathematics Honor Society

COMPUTER EXPERIENCE

Languages: Java, C, C++, Visual Basic, x86 Assembly, PHP, XHTML
Operating Systems: Linux, Windows 9x/NT/2000/XP/Vista/7
Software Packages: Microsoft Office 98 – 07, Microsoft Visual Basic, Microsoft Visual C++, Macromedia Dreamweaver
Database Management: Oracle 9i, Microsoft SQL Server

RELEVANT EXPERIENCE

System Administrator, **Software Engineering I**, Loudonville, NY, September 2009 – Present

- Maintained Software Team's Workstation's systems and applications software
- Managed all user profiles on Team Workstations
- Managed preparation and management of Team documents

ADDITIONAL EXPERIENCE

Electrical Maintenance Extra, **The Metropolitan Opera House**, New York, NY, July 2009 – August 2009

- Maintained light fixtures in theatre and surrounding areas
- Replaced and cleaned every light bulb of each chandelier in both theatre and lobby areas
- Responded to miscellaneous office maintenance work orders

Cashier, **Star Wines and Liquor,** Monroe, NY, May 2008 – August 2008

- Developed strong communication skills handling difficult customers
- Unloaded delivery trucks/stocked shelves; maintained organization of warehouse
- Trusted to make deliveries to customer's residences

Seasonal Laborer, **Otterkill Golf and Country Club**, Campbell Hall, NY, May 2007 – August 2007

- Miscellaneous Course Maintenance
- Various landscaping projects around club house and restaurant
- Expected to complete various course maintenance projects independently
- Learned to work well as part of a team on larger projects

4.1.4 Christopher Hughto (Webmaster)

Christopher Hughto
8 Canvasback Ridge
Waterford, NY 12188
518.235.8550
cj12hugh@gmail.com

OBJECTIVE

To obtain a challenging position in the field of computer science.

EDUCATION

Siena College, Loudonville, NY
B.S. Computer Science, May 2010

COMPUTER EXPERIENCE

Languages: C++, Visual Basic, x86 Assembly, Java, PHP, XHTML,
Operating Systems: UNIX, Windows 95/XP/Vista
Software Packages: Microsoft Office 98 – 07, Microsoft Visual Basic, Microsoft Visual C++, Macromedia Dreamweaver, Adobe Flash, Adobe Fireworks

RELEVANT EXPERIENCE

Webmaster, **Software Engineering I**, Loudonville, NY, September 2009 – Present

- In charge of programming team website.
- Maintains team website.

ADDITIONAL EXPERIENCE

Sales Associate, **The Shoe Dept,** Clifton Park, NY, August 2005 – September 2009
Student Worker, **Siena College,** Loudonville, NY, September 2008 – Present
Sales Associate, **Staples,** Latham, NY, September 2009 - Present

HONORS/AWARDS

Second Degree Black Belt, **U.S. Budokai Karate Association,** Clifton Park, NY

- Taught classes to younger students
- One of the highest ranks obtainable in a martial art.

4.1.5 Erik Stegmann (Webmaster)

Erik J. Stegmann
89 Fairway Drive
Valatie New York 12184
(518) 469-7963
erstegmann@yahoo.com

Education
Siena College, Loudonville NY
B.S. Computer Science, May 2010
Minor: Multimedia Development

Computing and Technology experience
Languages: Visual Basic, Java, C++, x86 Assembly Language, PHP, MySQL, CSS, XHTML, AJAX
Operating Systems: Windows XP/Vista, Mac OSX.3/OSX.4/OSX.5
Software Packages: Microsoft Visual Studio, Macromedia Creative Suite, iLife, Adobe Photoshop Microsoft Office Suite, X Code, iPhone Development Package, Microsoft Project
Bug Tracking Software: Bugzilla, Devtrack, XBwatson
Source Control Systems: Perforce, CVS (Concurrent Version System)
Debugging Hardware: Xbox 360 Test kits Playstation 3 Test kit, Playstation 2 Test kit, Playstation 2 Dev kit, Playstation Portable Dev kit, Nintendo Wii Reader, Nintendo Wii Hardware kit, Nintendo NDev, Nintendo Nitro DS.

Relevant Experience
Quality Assurance Testing Engineer (Internship and Employee) Vicarious Visions Inc., An Activision-Blizzard Company, Menands, NY September 2004 – Present

- Member of a QA Team which tested and identified code and quality defects of numerous video game titles.
- Provide design input to project teams involving game play, weapons, hit points, and controller layout.
- Tested on platforms from Microsoft, Nintendo of America and Sony Computer Entertainment of America.
- Identify, Enter, Verify, and Track software defects in bug tracking software systems.
- QA Team member for million selling titles, such as Doom 3 for Xbox, Guitar Hero World Tour.
- Completed games for the first billion dollar videogame franchise, Guitar Hero.
- Compiled data to be submitted to strategy guide publishers.

Graphics and Interfaces Designer, Siena College Loudonville, NY, September 2009 – Present

- Designed a dynamic team website
- Created an intuitive and attractive layout
- Consistently updated the website to reflect changes in our project

Member of the iPhone development community

- Working on independent projects designing useful and or entertaining apps for iPhone users

Dale Carnegie Graduate

- Attained useful skills for leading a team to accomplish a goal
- Developed communication, public speaking, motivation and stress relieving skills

4.2 (Appendix Continued) Glossary of Terms

The following are a list of technical terms used within the document. This section is provided to clarify their meaning.

AJAX: (Asynchronous JavaScript and XML) is a group of interrelated web development techniques used on the client-side to create interactive web applications. With AJAX, web applications can retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page.

PHP: PHP Hypertext Processor is a programming language used to create dynamic web sites. Has the ability to interact with a database.

HTML: Hypertext Markup Language is the scripting language used to describe the information contained on a website. HTML utilizes Cascading Style Sheets (CSS) to generate the style of the page. HTML and CSS are parsed by web browsers, such as Internet Explorer and Firefox, to render the websites for users.

MySQL: A free implementation of a Relational Database Management System. Used to store and retrieve information relevant to the website, such as usernames, passwords, problems, solutions, and scores. Accessing information within the database is achieved by submitting a "query" in the Structured Query Language (SQL) form.

CSS: Cascading Style Sheets are a simple way to add style to web documents.

Java: A programming language which the System will be able to compile and execute. This language will be used by the students to solve the assigned problems.

Java SDK: Software Development Kit for Java – a collection of tools used by developers to aid in the creation of programs.

Java Runtime Environment: A collection of programs used by Java to run Java programs.

JOLT: Java Online Learning Toolkit is the name of the proposed system that *518 Interactive* plans to develop.

Hardware: The tangible components of a computer and server. Examples include monitors, disk drives, printers, keyboard, processor, and memory.

Software: The intangible components of a computer and server. It is a set of machine-level instructions that is run from within the memory, and is used to perform a specific set of functions. Examples include Microsoft Word, Adobe Photoshop, and Mozilla Firefox.

This page intentionally left blank

End of Software Plan