

# **CSITschools**

## **Interactive Web-based Learning Modules for Beginning Computer Science Students**

Luke Pederson & Drew Callan  
May 2013

CST400 Capstone Planning  
Advisor: Kate Lockwood  
Instructor: Kevin Cahill  
California State University Monterey Bay  
School of Information Technology and Communication Design

# Table of Contents

Title Page .....	1
Table of Contents .....	2
Abstract .....	3
Executive Summary .....	4
Introduction/Problem Description .....	5
• Background & Problem Description	
• Target audience	
• Environmental Scan	
Solution Description .....	6
• Description	
• Other possible solutions	
Methodology / Implementation Plan .....	7
• Resources	
• Risk analysis	
Changes .....	8
Deliverables .....	9
Testing and Evaluation Plan .....	10
• Functional testing	
• Usability testing	
• Evaluation	
Budget .....	11
Timeline .....	12
Appendixes .....	13

# Abstract

CSITSchools is a website that teaches students Computer Science concepts. It is a free alternative to traditional textbooks. It provides students with resources like syntax examples, visual queues and executable code. The visual elements and browser executable (details further in document) enhance student interest and engagement. The modules within the site will be logically structured to correlate with the pathways defined by CSIT faculty members. The final deliverable of the project will be used as part of the Computer Science and Information Technology department and will specifically supplement the courses CST 231 and CST 238.

# Executive Summary

There is a growing demand among students, faculty, and administrators to adopt cheaper, more effective means of access to information. More specifically, the exponential growth of communication technology is demanding that in order to keep up with these rapidly changing times, the development of interactive, electronic alternatives to textbooks must be made a priority. By shifting away from these fixed, static sources of information and replacing them with more visually oriented, interactive media, a student's understanding of the material will greatly increase. Additionally, it also helps alleviate the burden on faculty members which arises out of trying to teach each student the foundational concepts of programming.

Various classes offered in the CSU Monterey Bay in the Computer Science and Information Technology (CSIT) department, have completely removed the traditional practice of buying and teaching with textbooks. They now instead utilize electronic materials such as online websites, videos, applets, etc... In these frequently changing times, a new, more effective means of providing information regarding the concepts of Computer Science to beginning CSIT students is required.

The immediate solution to this issue is for the CSIT department at California State University Monterey Bay to begin developing interactive learning modules that will both teach and demonstrate the basic core concepts of the various Computer Science courses. They will present information to students in a way that will ultimately help them more effectively learn the primary concepts of whatever field they choose to specialize in. The pathways outlined in each module will be strategically designed to compound student learning capabilities, student interest in those topics and the ability to foster a climate of individual growth by providing personalized interaction with the course material. The modules will specifically cover basic concepts such as Data Types and Conditional Statements as well as more advanced topics such as Pointers, Data Structures, and Recursion.

# Introduction

## Background & Problem Description

Technology is continually becoming more embedded in our lifestyles. The shift from physical to electronic mediums is more apparent, including our education system. This shift had led to lower costs for students, faculty, and the overall education system. Budget cuts over the last few years have led to innovative alternatives to traditional curriculums. Of these alternatives, many of them are electronic based.

Students have a growing preference towards using electronics resources as classroom materials. Traditionally, faculty and students are forced to buy textbooks that are expensive become out-of-date very quickly as new versions of the textbook are released. Students can spend more than \$1,000 dollars annually on textbooks that may only last one semester. Using electronic based alternatives allows for new versions to be distributed without creating new costs for faculty and students. They promote longevity since they aren't static and can be easily updated when course curriculums or individual classes need adjustments. Our client was searching for an effective learning tool to meet this growing preference.

## Target Audience

The audience of this project was future Computer Science teachers and students at CSUMB. Specifically, faculty and students enrolled in CST 231 and CST 238 would be the first to use our finished project as an essential resource for the class. Successful deployment and positive response to this project as part of the CSIT curriculum will demonstrate to other departments at CSUMB, and possibly other universities, that transitioning to electronic based learning tools is a viable alternative to traditional textbooks.

## Environmental Scan

A website that our project will resemble is W3Schools.com. It is a site designed to teach users about programming in HTML, CSS, Javascript, PHP, etc. The site provides documentation for all the skills needed to program for the web. It provides text documentation for proper syntax,

links to other valuable resources, and even interactive forms to test in real time Javascript code blocks. Instead of web programming, our project teaches users about C++ programming for software development. It also has media elements like videos to help demonstrate Computer Science concepts in an understandable way. Other sites like W3Schools exist, but this is the best example of a website similar to ours.

## **Solution**

### **Description**

This project provides an effective learning tool that faculty can use teach students the core concepts of programming. It provides a low cost alternative to the current curriculum materials for CST classes. It also has the capacity to be maintained and altered for future use without costs to students or faculty. This will stop instructors and students from having to frequently buy new versions of textbooks.

The goal is future implementation of the learning tools in the CSIT department of CSUMB. Positive response from faculty and students will be a key goal as well. It should be able to be maintained for longevity in the CST department's curriculum. Long term success of the project will demonstrate to other departments that use of electronic alternatives can be cost effective and still meet curriculum standards.

This project came into existence from our client's transition from physical documents to electronic. Our client needed a viable electronic alternative. Upon successful completion, the final deliverable will be used by our client in their curriculum for CST classes. It will meet all the standards defined by the university for the curriculum of the class.

### **Other Solutions**

Other possible solutions can be described as subsets of the goal of our project. There currently aren't any widely established interactive C++ teaching websites. There are many text-based websites that provide valuable documentation. However, their raw presentation does not promote easy learning for beginning computer science students. Different solutions could include different mediums for presenting the material in an easy to understand format.

Our project will differ from some of these other sources because it will incorporate all of their concepts presented, but present them in innovative and easy to understand format. It will package the the concepts and display them in different ways like text, images and browser-executable programs.

## **Methodology**

### **Implementation Plan**

This project was managed by both Luke and Drew, but the workload was divided between us. Each of us worked on sections of the project we were interested in, or had more skill in. Drew, having a greater interest in the mechanisms driving the underlying code, focused on developing robust programs and scripts that are able to handle the potentially dangerous input from users. Luke focused on the website interface to promote easy understanding of the C++ concepts. Connection of the website and back-end coding required our combined efforts to create a fast and reliable connection. Managing the project from start to finish also took our combined efforts. Since we both worked independently and dependently, we needed constant communication to maximize our time and effort.

## **Resources**

The resources for this project included everything required for a fully operational web server, personnel costs, and other required supplies. Labor required the use of a computer to carry out development and debugging of code to be placed onto web server. While in development, we also focused on making clear documentation and using maintainable coding practices. Creating documentation of coding will provide capacity for maintenance and longevity of project.

The required knowledge to carry out this project has been primarily gained from CSIT courses taken at CSUMB. Courses like CST 231 and 238 taught us the concepts of C++ that we will be presenting. Courses like CST 336 and CST 201 provided us with the ability to create meaningful website designs with interactive web pages. The CSIT curriculum gave us all programming skills we needed to successfully complete this capstone project.

Additionally, the faculty and instructors of the CST 231 and CST 238 classes helped us develop the curriculum that will be integrated into the modules. Specifically, Prof. Kate Lockwood and Dr. Young-Joon Byun, both have taught the classes with academic success. As they have already designed and taught many previous classes, we worked with them to more effectively design our modules. By working with them to help structure our web pages it is our hope that the pages convey the material they would like to teach and it is set up in a way that allows them to accurately and efficiently utilize our modules during their course lectures.

## Risk analysis

There were several events that could have caused the project to not be completed on time. Debugging and modification of server-side code took time to ensure it was robust. User input was the most risky part of the interactive process. Modifications to the structure of the website or interactive modules to maximize student learning also took time; unclear presentation of Computer Science concepts could have caused a major delay. Other stalls to the project could have include hardware problems, lack of software accessibility and documentation, or other unforeseen problems.

## Changes

Throughout the course of the project the goal of the project has remained the same, but a few changes have happened to the implementation. Problems with hardware and server permissions forced us to switch servers for development and deployment. Also various time and project constraints limited our ability to completely include all the features we had originally planned. While we did include all the aspects we were originally planning to, they may not be as completed as we had hoped. Specifically, we are talking about the visual aids that we used. We were limited in providing interactivity we could include because the Javascript library we used - KineticJS - was a developmental package and provided low-level function calls which made creating the visual elements tedious and time consuming. In addition, the roles of Luke and Drew switched during the project. Luke developed the back-end code and Drew focused on the front-end.



# Deliverables

Completion of this project produced a website with multiple layers. Students visiting the site needed to be able to easily understand the information presented. The project was composed of 2 primary layers. The top was the presentation layer, the website. It was designed for ease of use and understanding. The second layer was the underlying code that interprets, analyzes, and executes the students' input.

## **Website - User Interface**

The website is composed of styled HTML using CSS which hosts text, pictures, and interactive media. The pages are structurally designed to allow students to follow learning pathways to maximize learning. The learning pathways are comprised of modules that present individual concepts of C++ programming. The modules contain descriptions of concept, documentation of syntax, and various media elements. The key media element of a learning module are interactive forms that take user input of C++ code, send it to the server for compilation and execution, and return it to the browser.

## **Server - Framework**

The host server is designed to allow easy maintenance of the website and application framework. The framework will consist of a Unix-based operating system that will contain software to compile and execute C/C++ code. PHP will gather input from users and send it to server. It will also grab the C++ code execution output and compose structured HTML pages to present to the user.

## **Documentation**

Students are offered documentation on the website to help guide them through usage of learning modules and pathways. Faculty are offered documentation to help them customize modules for curriculum changes. Administrators are given documentation explaining server and project structure for easy maintenance and modification of application framework or website structure.

# Testing and Evaluation Plan

## Testing

Functionality testing for our capstone involved debugging and modifying the server-side code for accurate display of execution results to website. Input gathered from forms on the website is passed to a server-side C/C++ compiler, executed, and returned to the browser. Our testing was divided into steps that verified that all steps of this process are working as expected. The most dangerous step of the process was gathering user input and passing it to the compiler; the input will have to undergo validation checks. Functionality testing and analysis will be divided into the following portions of the project.

- Server-side compilation and execution
- Website I/O for interactive modules
- Website design structure

Usability testing for our project was also a primary concern. Our project had to be easily navigated and understood by students. Surveys were conducted with a test group of students to receive feedback about the site; modifications were made accordingly for maximum usability. Maximizing the effectiveness of learning modules was an ongoing process since it cannot be perfected, only improved. Management of the site will also need to be easily done by faculty members since curriculum requirements may change from year to year. The site must be able to accommodate these changes. Demonstrations and testing for various faculty members was conducted to increase flexibility and longevity of the website by gathering feedback and modifying the website accordingly.

## Evaluation plan

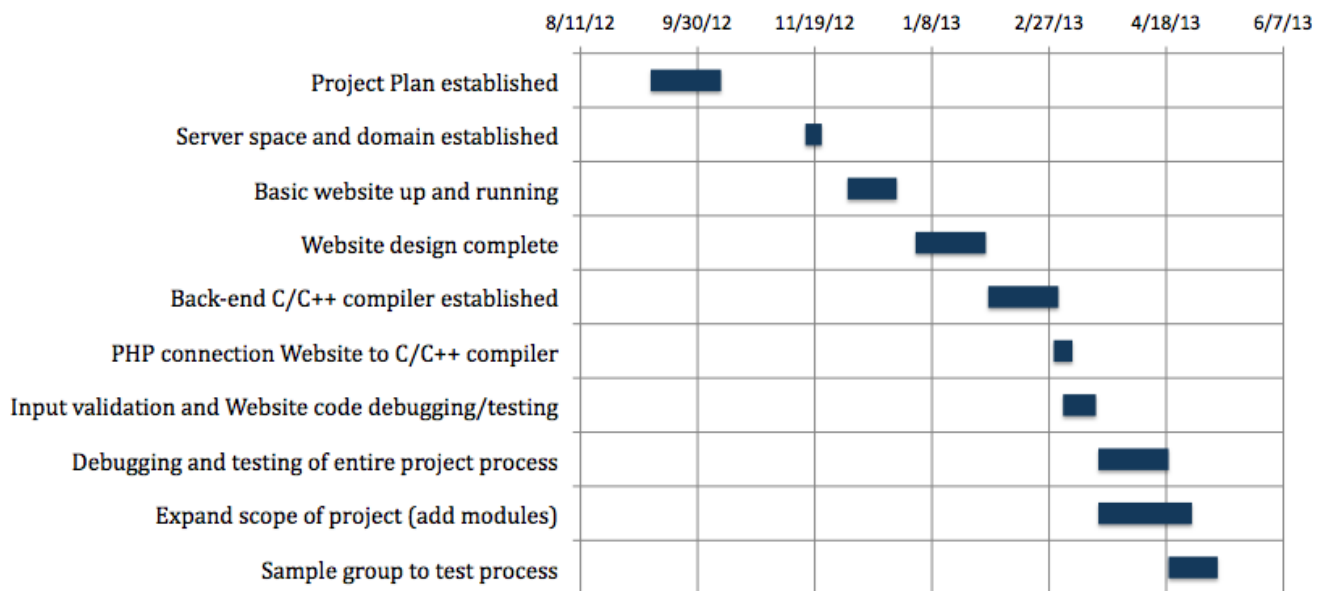
The project will be successful if it meets all of the criteria of our client. Our client will be given demonstrations of the project as it progresses so the project plan and goal may change with the client's needs. Positive response from students is also required for this project to be a success, since they will be the primary users. The most optimal outcome for the project is if it is used by the client as active part of the curriculum for the CSIT department.

# Budget

This is an estimated budget plan for our project if it were to be completed by established professionals in the market. Domain name costs range from around \$50 to \$100 dollars for the first 2 years and drop in price thereafter. Buying a prebuilt server can range drastically in price depending on the specifications of the system. Systems that are expected to have high traffic will cost much more than a low-end system. Again, Internet Service Provider costs will have a large range of costs depending on the amount of traffic that is expected. These estimates are based on a small business connection using a T1 or T3 connection. Labor costs were found from pay scale statistics from the industry. These are estimated costs for completion of the project from start to finish.

	<u><b>Costs</b></u>	<u><b>Units</b></u>	<u><b>Estimated</b></u>	<u><b>Actual</b></u>
Domain	\$100	2 yrs	\$100	\$0
Server	\$700	1	\$700	\$0
Software	\$100	1	\$100	\$0
ISP - Internet	\$1400	2 yrs	\$2800	\$0
Labor	\$40/hr	150 hrs	\$6000	\$0
Computer	\$800	1	\$800	\$0
<b>Totals</b>	~	~	<b>\$10,500</b>	<b>\$0</b>

# Timeline



# Appendixes

## PowerPoint

### Interactive Learning Modules for CST Students

Drew Callan & Luke Pederson  
CST400 Capstone Planning Fall 2012  
Cal State Monterey Bay  
School of Information Technology and Communication Design  
Capstone Concept Proposal  
Client: Kate Lockwood & CSIT Dept.  
Capstone Advisor: Kate Lockwood  
21 September 2012

### Recommended Solution

- Design, Create, and Implement web-based modules to showcase basic concepts of C++ programming to students
- Each module allows for students to interactively practice coding, debugging, and executing basic C++ programs
- Each module will focus on a singular core C++ concept to showcase their significance to students
  - Variables, Arrays, Pointers, etc...

### Skillsets & Resources

- Additional Skillsets Needed
  - Deeper understanding of HTML5 & JavaScript's capabilities
  - Designing & Implementing webpage layout
- Additional Resources Needed
  - Domain Name
  - Web Host
  - Media Editing Software

### Problem Statement

**Problem:** Students often have difficulty understanding the core concepts of software development

#### **Project Background:**

- Typical study materials are often expensive and static
- CST 231 & 238 wish to embrace the shifting trend of moving away from traditional study materials/methods
- Alternative study and classroom materials required to adapt to the emerging needs of both students & instructors

### Project Deliverables

- Help/Troubleshooting guides and other text based documentation for students and instructors
- Interactive capabilities and tutorial videos, to demonstrating the usage of core programming concepts
- Organized modules for efficient learning and progressing understanding of key programming concepts
- Final deliverable will have capacity to be used by CST Instructors as a primary teaching resource at CSUMB

## Reference List

### Budget References

Domain - <http://www.allbusiness.com/technology/internet-domain-names/>

Server - Pcmag.com

ISP - Buyerzone.com

Labor - Payscale.com

## Resume(s)

Available upon request.

## Copyright permissions

© 2013 Luke Pederson and Drew Callan

## Collaboration Statement

We would like to thank our client and advisor Professor Kate Lockwood for all her help and encouragement throughout the course of this project. We would also like to thank Kevin Cahill for his guidance as well as the CSIT faculty and department for their support. We'd like to express our congratulations for the graduating classes from both CSIT and CD for Spring 2013. We did it.

## Final Documentation

Please see attached disk (CD) for project documentation.