Project Plan

nSite Central Software Suite

developed by
Team DunKyan

Michael Dunn
Kyle Kerrigan
Ryan Sessions

Version 2.0

Table of Contents

Introduction

This document outlines the strategy to which Team DunKyan will adhere when developing the nSite Central Software Suite. Included in this document is the statement of work, a list of all resources required to complete the project, the roles each member in the team will play, the schedule of development, and an enumeration of potential risks the project might face.

Statement of Work

We have been asked by our client, Patterson Dental, to develop software that will help in the collection and centralization of data. To meet this request, we are developing nSite Central, which is a product suite allowing a corporate office to connect, manage, view and report against any number of nSite Central or straight EagleSoft installations. One portion of this software package is web-based, allowing corporate users to access and view information from client locations, while the other portion is an agent-based data mining system responsible for the collection and retrieval of pertinent information.

Resource List

In order to complete the project in a timely manner, the following resources will be required:

- at least 2 computers running Windows XP or higher (for testing purposes)
- an IDE for web development (Microsoft Visual Studio 2008)
- web application development framework (ASP.NET)
- database software (MS SQL Server 2008)
- two different web browsers in which to test our code (IE7 and FX3)
- version control software (Subversion)
- web server software (IIS)
- books and other materials necessary for learning ASP.NET and MS SQL
    - C# and WCF if time permits for agent development
- a conference room to hold meetings and reviews
- email, phone, and/or live video services by which meetings with client can take place

Roles

This project will require many different types of programming knowledge and expertise; therefore, in order to better define the development of the project, each unique role is defined below. It is important to note that since our team consists of only three members, and since we plan on doing pair programming in certain development situations, there may be one person or multiple people assigned to each role.

Role	Team Leader
Member	Michael Dunn
Description	The person in this role is responsible for managing the team, maintaining contact with the client, making any crucial decisions that have a major impact on the project, maintaining the team website, and managing all important documents.

Role	Researcher
Member	Michael Dunn, Kyle Kerrigan, Ryan Sessions
Description	The person in this role is responsible for seeking out resources and materials necessary for learning about more technical aspects of the project. For example, one in this role would have to acquire specific books about ASP.NET in order to learn how to use it, and it is also the responsibility of the person in this role to share any knowledge they gain with all other members of the team.

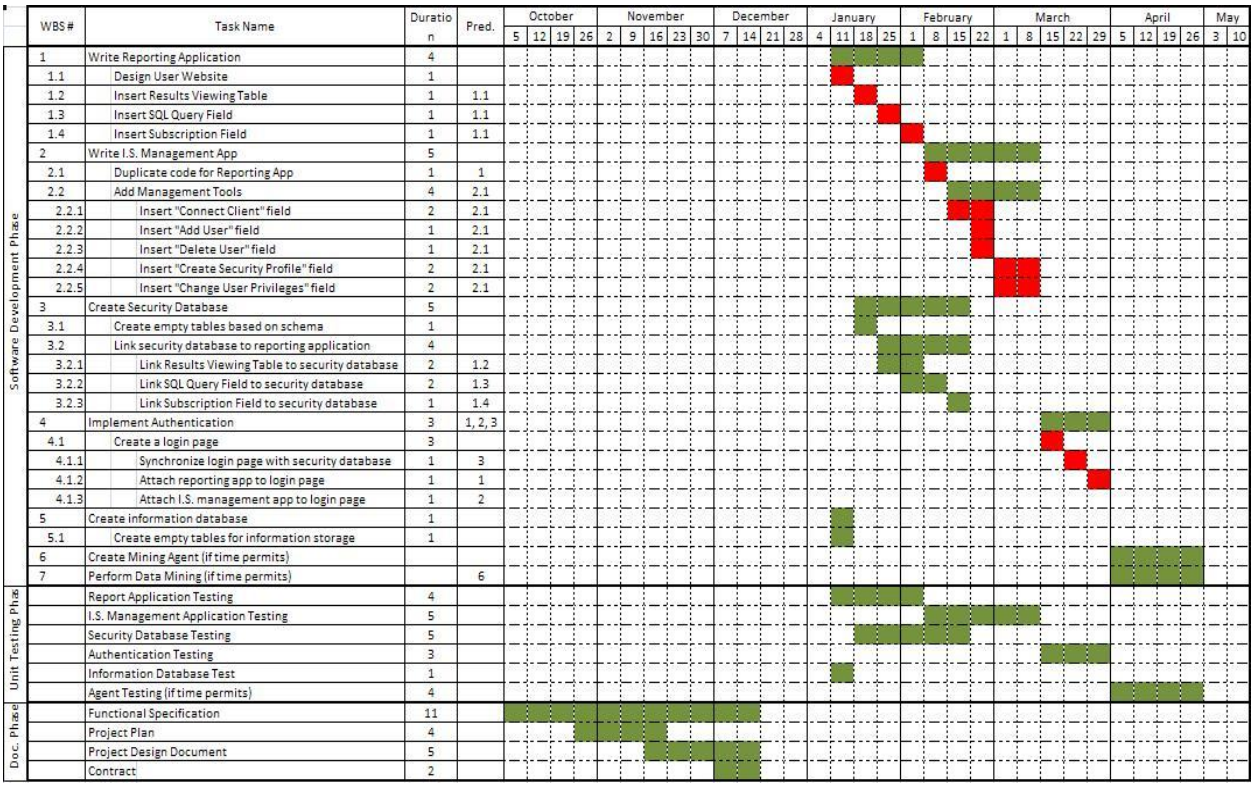
Role	Interface Designer
Member	Kyle Kerrigan
Description	The person in this role is responsible for designing the "look and feel" of the graphical user interface of the Reporting Application and the I.S. Management Application. This person will design the layout of every single page in both systems.

Role	Database Specialist
Member	Michael Dunn
Description	The person in this role is responsible for implementing the schemas for both the security database and the data storage database. If time allows, this person will implement logic that allows the user to customize SQL queries.

Role	Network Designer
Member	Ryan Sessions
Description	The person in this role is responsible for the logic to establish and maintain connections between the main application and client sites. If time allows, this person will be responsible for the design and implementation of the data mining agent system.

Schedule

Below is a Gantt chart outlining the major steps in the development process for nSite Central. The chart contains the name of each task, the projected time it will take to complete each task, a list of preconditions (i.e. other tasks) that must be completed prior to the start of each task, and a schedule that shows specific dates during which each task will be worked on and completed. The critical path is shown in red. Note that this is a working chart, and therefore is subject to change upon review.



	WBS #	Task Name	Duration	Pred.
Software Development Phase	1	Write Reporting Application	4	
	1.1	Design User Website	1	
	1.2	Insert Results Viewing Table	1	1.1
	1.3	Insert SQL Query Field	1	1.1
	1.4	Insert Subscription Field	1	1.1
	2	Write I.S. Management App	5	
	2.1	Duplicate code for Reporting App	1	1
	2.2	Add Management Tools	4	2.1
	2.2.1	Insert "Connect Client" field	2	2.1
	2.2.2	Insert "Add User" field	1	2.1
	2.2.3	Insert "Delete User" field	1	2.1
	2.2.4	Insert "Create Security Profile" field	2	2.1
	2.2.5	Insert "Change User Privileges" field	2	2.1
	3	Create Security Database	5	
	3.1	Create empty tables based on schema	1	
	3.2	Link security database to reporting application	4	
	3.2.1	Link Results Viewing Table to security database	2	1.2
	3.2.2	Link SQL Query Field to security database	2	1.3
	3.2.3	Link Subscription Field to security database	1	1.4
	4	Implement Authentication	3	1, 2, 3
	4.1	Create a login page	3	
	4.1.1	Synchronize login page with security database	1	3
	4.1.2	Attach reporting app to login page	1	1
	4.1.3	Attach I.S. management app to login page	1	2
	5	Create information database	1	
	5.1	Create empty tables for information storage	1	
	6	Create Mining Agent (if time permits)		
	7	Perform Data Mining (if time permits)		6
Unit Testing Phase		Report Application Testing	4	
		I.S. Management Application Testing	5	
		Security Database Testing	5	
		Authentication Testing	3	
		Information Database Test	1	
		Agent Testing (if time permits)	4	
Doc. Phase		Functional Specification	11	
		Project Plan	4	
		Project Design Document	5	
		Contract	2	

October: 5, 12, 19, 26
November: 2, 9, 16, 23, 30
December: 7, 14, 21, 28
January: 4, 11, 18, 25
February: 1, 8, 15, 22
March: 1, 8, 15, 22, 29
April: 5, 12, 19, 26
May: 3, 10

Risks

As with any project of this magnitude, there are many associated risks we cannot ignore. Thus, by acknowledging them now and defining ways in which to counteract them, our project plan may run more smoothly, effectively, and with a greater probability of avoiding the risks defined below.

Risk	Probability	Impact	Actions
Inexperience with ASP.NET	100%	Since this is the main development language for our project, not knowing ASP.NET would effectively halt all progress.	Assign each Researcher a specific topic in ASP.NET and have each Researcher share his findings with the rest of the team. Do individual research.
Inexperience with MS SQL Server	100%	Since this is the only database language we will use for our project, for both information storage and security purposes, not knowing MS SQL Server would disrupt our backend database setup, effectively halting all progress.	Assign each Researcher a specific topic in MS SQL Server and have each Researcher share his findings with the rest of the team. Do individual research.
Team availability	40%	With our conflicting schedules, if we cannot find time to meet for review and development sessions, lack of communication will ensue and project will get off schedule.	Make time in mornings, evenings, or whenever free time is available for team to meet. Assign Team Leader task of coordinating the meeting times based on each member's schedule.
Acquisition of necessary resources	20%	If we fail to acquire the essential materials and tools, we will not be able to begin work on the project, and when work actually begins, it will be dictated by the tools we have at hand.	Do as much research as possible before coding process begins in the Spring. Make sure proper authorities who have access to required materials are aware of our need for them in advance.