Team A

Project Plan

CIS-470 – Senior Project



Table of Contents

[System Overview 2](#_Toc448588020)

[Project Development Management 2](#_Toc448588021)

[Organization and Resources 3](#_Toc448588022)

[Role(s) and Responsibilities 3](#_Toc448588023)

[Development Technologies 3](#_Toc448588024)

[Personnel 4](#_Toc448588025)

[Schedule & Milestones 5](#_Toc448588026)

[Delivery Milestones & Baselines 5](#_Toc448588027)

[Risk Analysis 6](#_Toc448588028)

[Software Engineering 7](#_Toc448588029)

[Standards and Procedures 7](#_Toc448588030)

[Development Methodology 7](#_Toc448588031)

[Development Resources 8](#_Toc448588032)

[Testing Procedures 8](#_Toc448588033)

[Configuration Management 8](#_Toc448588034)

Project Plan

|  |  |
| --- | --- |
| Team: | Team A |
| Team Members: | John Boley, Justin Byrne, James Coltman, Marshall Gibson |
| Date: | 04/11/2016 |
| Project Title: | Williams Specialty Company - - Business Automation |

# **System Overview**

Williams Specialty Company (WSC) wants to simplify (and improve) their workflow through the development (and implementation) of a proprietary Business Process Automation (BPA) application system. This system will include a host of business operations ranging from processing client and inventory data to generating vital business records and orders. Because this system is comprehensive in scope, the development should be approached as an overarching Business Process Management (BPM) strategy for WSC; maintaining and optimizing WSC’s core operational mechanisms such as business processes, reports, and data.

# **Project Development Management**

The Waterfall (or linear-sequential life-cycle) Model will be employed throughout the duration of this project. This Software Development Life-Cycle (SDLC) model will illustrate the (individual) developmental processes of this project in a linear sequential flow, implying that each developmental phase will only begin if it’s previous phase is complete; the result of one phase acts as the input for the next (sequential) phase:

Communication

Planning

Modeling

Deployment

Construction

**Project Initiation, Requirements Gathering**

**Estimating, Scheduling, Tracking**

**Analysis, Design**

**Code, Test**

**Delivery, Support**

# **Organization and Resources**

Team-members will be partitioned into four separate divisions to facilitate exclusive elements (or features) of this BPA application. These divisions, in conjunction with each team-member paired with these responsibilities, can be observed below:

# **Role(s) and Responsibilities**

|  |  |
| --- | --- |
| Division | Team-member(s) |
| Database Administration | ***John Boley, Joe Gibson*** |
| Backend Application-Development | ***Justin Byrne, James Coltman*** |
| Frontend Application-Development | ***Justin Byrne*** |
| Application Modeling & Documentation | ***Joe Gibson*** |

In addition, team-members, in accordance with their unique role(s) (and responsibilities) will be utilizing one (or more) of the below mentioned technologies throughout development:

# **Development Technologies**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Technology | Description | | | | | | | |
| *PHP* ***v5.6.19*** | | ***General Purpose Scripting Language; Hypertext Pre-Processor*** | | | | | | |
| *To developed applications core abstraction layer, logic, and operations* | | | | | | | | |
| *PHPUnit* ***v5.0.10*** | | | | | | ***Unit-testing Framework for PHP*** | | |
| *To unit-test individual application processes (or methods) to ensure their* | | | | | | | | |
| *PHPDocumentor2* ***v2.8.1*** | | | | | ***Application Program Interface (API) Documenter*** | | | |
| *To analyze (and parse) core application methods to generate API documentation* | | | | | | | | |
| *Apache* ***v2.4.18*** | | | | | | | | ***HTTP Web-Server*** |
| *To self-host and execute (individualized) aspects of this application* | | | | | | | | |
| *Xdebug* ***v2.2.5*** | | ***PHP Extension Providing Debugging and Profiling Capabilities*** | | | | | | |
| *To test, debug, and profile this BPA application as it advances through its developmental stage(s)* | | | | | | | | |
| *MariaDB* ***v10.1.10*** | | | ***MySQL Relational Database Management System (RDBMS)*** | | | | | |
| *To store vital business records (and data) pertinent to the operation(s) of this BPA application.* | | | | | | | | |
| *JavaScript* ***v1.8*** | | | | ***High-Level Dynamic Interpreted Scripting Language*** | | | | |
| *To create (and incorporate) various User-Interface (UI) behavioral functions for this applications front-end* | | | | | | | | |
| *jQuery* ***v1.12.1*** | | | | | | | ***Feature-Rich JavaScript Library*** | |
| *To easily implement various complex functions (i.e., animations, document manipulation, event handling, etc…)* | | | | | | | | |
| *CSS* ***v3*** | | | | | | ***Cascading Style-Sheet Language*** | | |
| *To stylize the presentation layer of parsed HTML elements* | | | | | | | | |

Communication between all team-members will be administered through a combination of weekly (scheduled) video conferences (via Skype), e-mail correspondence, discussion thread posts, and personal phone calls and/or text messages; throughout the duration of this project’s life-cycle.

# **Personnel**

The below section defines each team-member’s role(s) and responsibilities:

|  |  |
| --- | --- |
| Name | Role(s) |
| *Justin Byrne* | **Project Director & Lead Application-Developer** |
| Responsible for overseeing and aiding in the developmental process of this application’s core architecture and internal elements, while supplying (individual) team-members with vital data, software packages, and tools relating to the development process. Organizing (and facilitating) team meetings to ensure that all vital business and/or application information is properly communicated across all team-members, while (continuously) maintaining a high-level of moral amongst team-mates throughout this application’s life-cycle. | |
| *Joe Gibson* | **Application Modeling Engineer & Documenter; Co-Database Administrator** |
| Responsible for modeling (or diagraming) this application’s framework, ensuring all API documentation properly meets application specifications (and standards), as well as any other related documentation, diagrams, and/or materials. Will also aid in diagramming (or illustrating) this application’s RDBMS alongside the lead DB administrator; John Boley. | |
| *John Boley* | **Database Developer & Administrator** |
| Responsible for designing RDBMS elements (tables, views, queries, indexes, etc…), and working in development and testing environments to ensure data flow is accurate to system requirements for this application. Will also work alongside App-Modeling Eng. to develop an accurate model (or illustration) of this applications database schema; as well as any other affiliated documentation. | |
| *James Coltman* | **Backend Application Developer** |
| Responsible for developing the core elements (classes, logic, data-types, etc…) intended on operating throughout this application’s back-end architecture; outlined throughout this applications system requirements. Developer will also be responsible for unit-testing all essential subroutines (or methods) to ensure accurate system results, while (routinely) profiling application to diagnose any (unnecessary) bottlenecks. | |
| *Justin Byrne* | **Frontend Application Developer** |
| Responsible for developing (and programming) vital front-end code utilizing a mixture of HTML, CSS3, JavaScript (JS), and jQuery (JQ) to create dynamic (and interactive) components for this BPA application’s client-side User-Interface (UI). Will also be responsible for bridging various data elements from the back-end architecture to the front-end UI. | |

# **Schedule & Milestones**

This section includes vital (or course critical) team milestones, including weekly course critical assignments and their subsequent due-dates.

|  |  |  |
| --- | --- | --- |
| Scheduled Job | Due Week | Due Date(s) |
| Project Plan | Week 1 | 03/06/2016 |
| Team Charter | Week 2 | 03/13/2016 |
| Requirements Specification | Week 2 | 03/13/2016 |
| Design Specification | Week 4 | 03/27/2016 |
| Component/Unit Code | Week 6 | 04/10/2016 |
| Test-Plan | Week 6 | 04/10/2016 |
| Deployment Plan | Week 8 | 04/17/2016 |
| Peer Review(s) | Week 8 | 04/17/2016 |
| Project Presentation | Week 8 | 04/17/2016 |
| Application Files | Week 8 | 04/17/2016 |
| Status Report(s) | Weekly (1-7) | 3/06 – 4/17 |
| Time Sheet(s) | Weekly (1-7) | 3/06 – 4/17 |

# **Delivery Milestones & Baselines**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task ID** | **Activity Name** | **Assigned** | **Start Date** | **End Date** |
| 1 | System Analysis |  | 02/28/2016 | 03/13/2016 |
| 1.1 | Project Plan |  | 02/28/2016 | 03/06/2016 |
| 1.2 | Team Charter |  | 03/06/2016 | 03/13/2016 |
| 1.3 | Requirements Specifications |  | 03/06/2016 | 03/13/2016 |
| 1.3 | Process Models |  | 03/06/2016 | 03/13/2016 |
| 2 | System Design |  | 03/13/2016 | 03/27/2016 |
| 2.1 | Database Design |  | 03/13/2016 | 03/20/2016 |
| 2.2 | I/O Prototype |  | 03/13/2016 | 03/20/2016 |
| 2.3 | Design Specification |  | 03/20/2016 | 03/27/2016 |
| 3 | System Development |  | 03/27/2016 | 04/03/2016 |
| 3.1 | Business Objects |  | 03/27/2016 | 04/03/2016 |
| 3.2 | User-Interface |  | 03/27/2016 | 04/03/2016 |
| 3.3 | API Documentation |  | 03/27/2016 | 04/03/2016 |
| 4 | Testing |  | 04/03/2016 | 04/10/2016 |
| 4.1 | Test Plan |  | 04/03/2016 | 04/10/2016 |
| 4.2 | Unit-Testing |  | 04/03/2016 | 04/10/2016 |
| 4.3 | Profiling |  | 04/03/2016 | 04/10/2016 |
| 4.4 | Component/Unit Code |  | 04/03/2016 | 04/10/2016 |
| 5 | Wrap-up |  | 04/10/2016 | 04/17/2016 |
| 5.1 | Deployment Plan |  | 04/10/2016 | 04/17/2016 |
| 5.2 | Project Presentation |  | 04/10/2016 | 04/17/2016 |
| 5.3 | Peer Review |  | 04/10/2016 | 04/17/2016 |

# **Risk Analysis**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Probability | Severity | Risk Score | Plan |
| General Lag; Time-table Conflicts | 4 | 4 | 16 | Try to re-distribute team-members’ tasks (and responsibilities) ASAP, to try to mitigate for any loss in time |
| Team Coordination/Cooperation | 2 | 4 | 8 | Continuously maintain an attentive awareness of each team-members overall level of participation |
| Advancing with imprecise guidance | 2 | 3 | 6 | Ensure that each team-member adequately understands their assigned roles/responsibilities throughout the entire direction of this project’s life-cycle |
| Documentation does not properly progress with project | 4 | 4 | 16 | Try to distribute remaining portions of incomplete documentation to various team-members, in an attempt to complete any (and all) missing portions |
| Magnitude of project not properly estimated | 3 | 6 | 18 | Try to appropriately scale the scope of the project down into manageable sections while (potentially) adjusting team-member’s roles (or responsibilities) to accommodate a new process paradigm |

# **Software Engineering**

This section will define the standards and procedures to which our team will adhere towards in order to develop a consistent product.

# **Standards and Procedures**

* All function (or method) written throughout this applications framework will include code appropriate comments, disclosing the function’s purpose, including the intended use of its parameters (or arguments) as well as the expected result(s) to be returned.
* To ensure continuity for all application code naming conventions, functions and variables will be formatted using underscores:

$variable\_one = “string”;

public function method\_name()

* To ensure continuity, database naming conventions for tables, rows, and indexes will follow camelCase standards:

tableOne

userID

* A similar (if not identical) naming scheme will be adhered to while importing and exporting data to (and from) the database, from the application.
* A data dictionary will be implemented to maintain a consistent database, while eliminating data anomalies and redundancies.

# **Development Methodology**

Our team will be adhering towards the Waterfall Model throughout the SDLC of this project, in conjunction with routine correspondence and collaboration to merge our strengths towards designing a coherent application model to comprehensively fulfill all system requirements. An illustration of the Waterfall SDLC model can be observed under Project Development Management.

# **Development Resources**

Each team-member will utilize their own (independent) Integrated Developer Environments (IDEs), ranging from Visual Studio (VS) to Sublime Text (ST), while working with XAMPP, a cross-platform web-server solution providing each team-member with the necessary PHP library files, MySQL RDBMS (MariaDB), as well as an Apache HTTP web-server to test (and troubleshoot) their assigned tasks; throughout this project’s developmental process.

For a more comprehensive list of this team’s developmental resources, please refer to the Development Technologies section under Organization and Resources.

# **Testing Procedures**

Testing will be facilitated through a series of coding techniques and third-party software. PHPUnit will be integrated into the Backend Developmental Workflow, to allow each Application Developer to (routinely) stress test their subroutines (or methods) as they are developed, while QUnit will be utilized for the Frontend JavaScript Developmental Workflow. After each team-member is self-assured that their work is satisfactory, they will submit their work to the Project Director (or team lead) for a final evaluation. Any unresolved issue will be discussed (and rectified) during weekly meeting; to ensure group collaboration.

# **Configuration Management**

Each team will be utilizing XAMP, which will include the majority of software packages and dependencies utilized throughout the duration of this project’s life-cycle. In addition, our team will be utilizing GitHub as an online (cloud-based) offsite repository, to host, share, and backup our project code as it evolves.