PROJECT MANAGEMENT PLAN

Project : IRI 1 -- Hydroviz

Team Members : Christopher Boudreaux, Tory Hebert, Phillip Huval, Jacob LeCoq, and Andrew Wise

Version Published : October 11, 2015

ABSTRACT

The purpose of this project is to design a web interface that allows a user to choose options from a different drop-down menus dealing with HydroViz data, then receive a visualization of the results in a chart generated by Highcharts. This plan lays out the steps we intend to take to accomplish this goal, in designing and implementing the front-end graphical user interface, the backend processing of the input retrieved from the HydroViz site, and the interfacing between our application and the Highcharts data visualization product. This document discusses team organization, life cycle model used, a risk analysis, resource requirements, the scheduling of our deliverables, monitoring and control mechanisms, and the professional standards that we hold ourselves and our work on this project to.

TABLE OF CONTENTS

LIST OF TABLES

INTRODUCTION

Introduction to the entire plan

The pages contained within this documentation contain a rundown of this project's assignment of roles to each team member, the reasons behind and details of the Scrum model that we have decided to follow, the potential risks that we believe could hinder our progress and our plans to deal with those risks should we come across them, our delivery roadmap, and an overview of how we intend to ensure our product is up to standards.

Purpose and Scope of the plan

This plan encompases the entirety of the project, and will aid in providing a solid foundation upon which to base the development of the product. It lays out the methods, tools, and organization of the project and rationalizes the choices made for each. This will allow the team to more readily refer to this information in the future should questions about how to conduct ourselves arise.

Brief Overview of the Product

The product is the internal functionality of a web interface on the HydroViz site that will allow the user to filter what data they will receive from the Hydroviz database. The filters, listed in order of user selection, include: Scenario, Extent, Time Period, Variable, and Data Type. The filters are all dependent upon their predecessors, so if the user were to change one of the values, the following ones will be reset so that the user can re-select an appropriate option. Once the user has selected the desired data, a button titled "Visualize Data" will appear, which, when selected, will allow the user to view the data in the form of a graph all without leaving the HydroViz site.

Description of the Structure of the Plan

As mentioned, the document is divided into 7 main segments: Project Organization, Lifecycle Model, Risk Analysis, Hardware and Software Resource Requirements, Deliverables Schedule, and Monitoring and Reporting. Each of these is further subdivided into at least two subsections: the topic mentioned in that section's name, and a rationalization of the choices described in that section.

PROJECT ORGANIZATION

The team working on the project will consist of five developers, each with their own specific tertiary responsibilities. Below is the listing of the team members, in alphabetical order, followed by their specified roles in the project and the rationale behind that member taking that role. We believe that the choice made for each team member represents the role that best exemplifies and exploits their personal strengths.

- Christopher Boudreaux
    - Role : Lead Designer and Architect
    - Rationale : Chris is good at high-level algorithmic and architectural design. He is good at seeing how poor decisions in system planning will cause issues down the line.

- Tory Hebert
    - Role : Lead Software Guru
    - Rationale : Tory currently works for LEDA and has the most experience out of any of us with the web languages we will be using on this project. He is quite talented with implementing new features rapidly and in a manner that facilitates later improvement.

- Phillip Huval
    - Role : Lead Test Guru
    - Rationale : Phillip worked as a full time tester for Centurylink, so he knows better than anyone what kind of tests need to be executed to maintain the security and reliability of the final working product.

- Jacob LeCoq
    - Role : Lead Project Management Guru
    - Rationale : Jacob has been a team leader for several other project-based courses in our curriculum, and has produced high-quality results in each.

- Andrew Wise
    - Role :Project Liaison
    - Rationale : Andrew is the most effective communicator of us all, therefore it makes sense for him to fill the role of Project Liaison.

LIFECYCLE MODEL USED

The Scrum agile lifecycle model will be used in the development of this product due to the client contact expressing that frequent meetings would not be necessary as long as our project liaison was in frequent contact with him, and that new tasks would be available periodically upon completion of the currently given task. He decided this because he believes it would more properly simulate performing actual contract work for a company, where smaller initial tasks may act as a way for the client to measure the competency of the contractor before entrusting them with the implementation of a crucial system. For these reasons, we have decided to set up our Scrum team as follows (These are somewhat titular, as we will all end up taking part in the implementation process) :

- Sprint length : 1 week

- Product Owner :
    - Jacob LeCoq - As project management guru, Jacob will already be in charge of prioritizing tasks for the team in the development process.

- Scrum Master :
    - Andrew Wise - As the project liaison, Andrew will already be the only one in direct contact with the client outside of the scheduled meetings. Therefore, the responsibility of effectively communicating the needs of the developers to the client and vice versa between meetings falls on him.

- Development Team Members :
    - Christopher Boudreaux
    - Tory Hebert
    - Phillip Huval

In addition to daily team meetings to discuss the current status and direction of project, we will hold our weekly Scrum meetings to discuss changes to the overall structure of the plan and analyze what we have learned from previous sprints in order to apply it to increases in future productivity. At the current time, we have tentatively scheduled another meeting with the client during the last week of October, and suspect that we will hold at least one more in November.

RISK ANALYSIS

- This section discusses potential roadblocks the project may hit, and briefly mentions why we believe this risk is a possibility, as well as how we plan to work around the risk should it become a reality. The format of this table is inspired by [2].

Table 1 : Risk Analysis

Risk	Likelihood / Rationale	Effect on Project	Contingency Plan
Team Member fails to perform duties (short term, < 1 week)	Moderate We all have additional work	Low	The load of that person's work will be spread across the available group members.
Team Member fails to perform duties (long term, > 1 week)	Low The unexpected can happen and cause trouble	High	That person will be confronted about the problem. If not rectified, issue will be formally reported to Dr. Kumar. or the TAs managing the course.
It is harder than expected to integrate Highcharts with the product	Moderate It is usually easier to plan an implementation than to execute it	Moderate	Effort will be diverted from other aspects to maintain the expected timeframe.
Client changes the functional requirements	Near-Certainty It is nearly unheard of for the requirements of custom software to remain static	Low	The implementation will be as modular as possible, so features are added easily.
The Design, Architectural, and Testing plans must be updated	High If the requirements change, additional infrastructure may be required to accommodate	Low	By the time these documents need to be changed, they will be completed. Only small changes will be necessary.
There are delays in the client sourcing further work after our completion of the first task	Moderate With the first segment of the project, there was some delay of decision between the contact and his superior	Low	The team is able to dedicate additional time to improving the documentation and optimizing the existing code.

HARDWARE AND SOFTWARE RESOURCE REQUIREMENTS

As we are only tasked with implementing a web page that parses a JSON object provided by a call to Hydroviz's provided api, it is not expected to be exceedingly expensive in terms of hardware usage. We suspect that even with minimal optimization of the code, the local hardware resources required to execute an operation in the program in its entirety, including an access to the Hydroviz site, parsing of the provided JSON object, and feeding that data into the Highcharts API will be nominal. Therefore, it is expected that the product will operate on most computers that are capable of running a modern browser.

Nonetheless, due to the expectation that this program must run on a wide variety of hardware configurations, it is necessary that we emphasize the need to take care in planning our algorithms and minimize the runtime complexity expenses of the system. Unfortunately, there are number of variables at play in a typical call to our system, so it will be impossible to have full control over the length of time it takes to provide results for a specified dataset, as this is largely dependent on the time required for Hydroviz's system to respond to our application's request.

The end user should not have to download or install any additional runtimes or platforms to run our code. All of the code should be executable fully within any modern web browser, and as such we must limit the tools we use to those that can either be embedded in a page or supported natively by browsers such as Firefox, Chrome, or IE. We will test the code in these three browsers, as they are the most widely used, but development and testing will likely focus specifically on Chrome, since that is the browser that most of the team members use frequently. As such, we will adhere to using only basic HTML, Javascript, and JQuery

DELIVERABLES SCHEDULE

- Sprint 1 : By the end of this sprint, we intend to deliver a working prototype of the user interface to the client. While this piece alone will not be incredibly functional, it will allow the client to offer input on what they think should be changed before we move on to the parts of the project that are more involved. The following steps will be required:
    - Design and implement the user interface
    - Complete the Project Management and Requirements documentation.

- Sprint 2 : By the end of this sprint, we intend to have delivered the finalized user interface along with the functionality of retrieving the JSON object from the client's site. The following steps will be required:
    - Designing and implementing the component that reads from the api
    - Complete the system architecture documentation

- Sprint 3 : By the end of this sprint, we intend to deliver the code that implements the highcharts integration. At this point, we intend to have a fairly clear idea of what the final product will look like. The following steps will be required:
    - Design and implement the component that passes information to the Highcharts library
    - Complete the detailed design documentation.

- Sprint 4 : By the end of this sprint, we intend to deliver the first full release of the product. This build will include, at a minimum, all of the functionality that was specified by the client and recorded in the requirements documentation. The following steps will be required after Sprint 3:
    - Design the component that retrieves and displays the result of the Highcharts functionality
    - Implement the component that retrieves and displays the result of the Highcharts functionality

- Sprint 5 onward : At this point, we will begin heavily refactoring the project to allow for ease of expansion and increased efficiency in order to accommodate any changing requirements given to us by the client. Additionally, the documentation will be amended as appropriate to ensure it is as accurate as possible.

MONITORING, REPORTING, AND CONTROL MECHANISMS

As mentioned above, the vast majority of our project monitoring will be spread across the daily meetings and the weekly end-of-sprint meetings. However, there are several other tools that will allow us to identify problems with the efficiency of various team members at a glance, and ensure that the project is proceeding as planned.

- Trello
    - Trello essentially offers a digital project board. This allows the team to subdivide goals into smaller, more manageable tasks, and split them up among group members according to their levels of skill. Additionally, it allows us to explicitly assign certain team members to cards, allowing us to pinpoint the reasons when certain aspects of the project are falling behind.

- Github
    - Github is a distributed version control system that enables a team to collaborate on software projects. As the name implies, it allows the team to keep personal copies of the project and make changes to those before integrating their changes into the main codebase. Once a change has been merged into the main repository, other team members are able to pull these changes into their local repositories. This has the added benefit of timestamping and allowing for commit messages, which serve as mini-changelogs. Additionally, it is possible to tell which team member submitted which changes, so poor practice can be detected and rectified.

- Google Drive
    - Google Drive allows for collaborative creation of documents, slide shows, diagrams, and spreadsheets. Specifically, for this project, the value in this service is in the ability to seamlessly integrate the types of documents together seamlessly. Real-time multi-user file editing is also a wonderful addition to the creation of software documentation, as it enables the team to divide the responsibility of preparing these documents and complete them in a far more rapid manner. Lastly, the service allows you to view previous versions of the documents, shows which users contributed to the document between versions, and highlights which users contributed which content.

PROFESSIONAL STANDARDS

The following provides a professional standards guideline that the team will follow. These guidelines have been agreed upon by each member in the team before the time of publication. The team has supplied our own self-tailored guidelines to follow along with those provided to us by [1]. Each team member shall act in accordance with the ACM / IEEE joint code of ethics as follows [3]:

1. Public : The Hydroviz website is a source of information for people making decisions related to Louisiana's waterways. This makes the public a stakeholder in the project.

2. Client and Employer / Profession / Product: Project members will hold themselves to the highest of standards as they work on the product. Our team acts as a representative of the level of competency produced by the university's computer science program, so poor performance reflects badly on not only ourselves, but also our colleagues and professors.

3. Judgment / Self: If a team member disagrees with some method the team is following, he will raise the concern with the team.

4. Colleagues: Team members will strictly adhere to the coding standards agreed upon by the team, including following the agreed upon formatting for blocks, using meaningful variable names, and creating a header comment for functions and classes, including a brief description of the functionality, the valid ranges for the parameters, and a description of the state of the system after returning from the function. Git commit messages will be descriptive of the commit itself.

5. Management: When the project management plan changes, team members shall adjust their activities accordingly.

REFERENCES

[1] Ashok Kumar, “Project Management Plan”, https://moodle.louisiana.edu/pluginfile.php/280791/mod_resource/content/1/Project-Management-Plan.pdf, 2015.

[2] Ian Sommerville, “Software Engineering, 10th Edition”, Pearson, 2011.

[3] “Software Engineering Code of Ethics and Professional Practice”, ACM / IEEE, 1999.