CSC 190 Section 01

Ashley Finger—Bai Xiong—Cody Lanier—Cody Prior—Daniel Gallegos—Michel Watson

03/19/2014

Software Project Management Plan

Behaving

Senior Design Project

Table of Contents

1. INTRODUCTION

This is the software system proposal document for the GotTrackDays.com project sponsored by the proprietor of Got Track Days Inc., Juan Calderon.

This project is being undertaken by the jBehaving development team. The team is comprised of undergraduate students majoring in Computer Science at California State University, Sacramento (CSUS). The team members are enrolled in a two-semester senior project course required of all undergraduate majors. Successful delivery of the desired software product will fulfill the senior project requirement for the student team members.

Project Sponsor:
Juan Calderon
Proprietor of Got Track Days, Inc.
Email: juan@GotTrackDays.com
Phone Number: (916)501-7147

jBehaving Development Team:

NAME	EMAIL	PHONE NUMBER
Ashley Finger	phyllangela@gmail.com	(408) 623-0062
Bai Xiong	bai.xiong91@gmail.com	(916) 583-4058
Cody Lanier	cody.lanier9@gmail.com	(916) 842-7869
Cody Prior	hiimcodas@gmail.com	(916) 740-0594
Daniel Gallegos	daniel.r.gallegos@gmail.com	(707) 803-1519
Michel Watson	michelwwatson@gmail.com	(916) 596-8253

Table 1 - jBehaving Contact Information

Team jBehaving set out to find an interesting and challenging senior project; one that would garner valuable industry experience in the field of Software Engineering. In acquiring sponsor, Juan Calderon, team jBehaving members ultimately secured a project that will fully meet their senior project needs. In order to assure that the project will be completed using the highest of engineering standards, jBehaving has constructed the following document as their Project Management Plan. The information issued by this document will explain exactly how jBehaving will be organized and how it will manage its workload. In order to confirm that the proposed management plan will facilitate the fulfillment of the sponsor's needs, the development team (jBehaving) must review and approve of said management plan.

This section of the Project Management Plan contains the purpose, scope, definitions, references, and overview.

1.1. PURPOSE

The purpose of this document, The Software Project Management Plan, is to inform the reader of how team jBehaving will guide the execution and the control of their project through the software development lifecycle. By reading this document, one will gain specific knowledge of how jBehaving will meet the sponsor's needs and success criteria.

1.2. SCOPE

This document will identify the project sponsor, the problem to be solved, the work to be performed, a timeline for completion, and an explanation of the general organization of the project. It will also contain general explanations of duties and expectations of team members at any given time during the project. Due to the potential for unforeseen factors, information in this document may be subject to change.

1.3. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

TERM	DEFINITION
Class	Template for creating objects
Charter document	The document which acts as an overview and overall contract of the project
Entity	Things or classes within software
Entity Relationship Diagram	A data model for describing the data or information aspects
Gantt Chart	A bar chart that shows the team's baseline schedule of the project
GitHub	Free internet repository for collaborative software development [4]
Gmail	Google's web email software
Google+	Pronounced: Google Plus, A social networking site [5]
Google Drive	Private, public, and/or shared web-based, cloud-storage location used to create and edit documents, spreadsheets, PowerPoint presentations, and more. A Google account is required. [6]
Google Hangout	A free text and video chat/instant messaging program. This is the group's primary means of communication outside of meetings [7]
Google Wallet	A payment system using debit cards, credit cards, loyalty cards and gift cards [8]

GotTrackDays.com	The website that Got Track Days Inc. will use to allow customers to register for track time [1]
Got Track Days Inc.	A business that allows its customers to register for track time at a variety of locations
HyperText Markup Language	The main markup language for creating web pages
Hypertext Preprocessor (PHP)	A server-side scripting language used for developing web pages
Integrated Development Environment	A software application that provides comprehensive facilities in software development
JavaScript	A dynamic computer programming language most commonly used for the development of web pages
jBehaving	The team and its members who will be working on the gotTrackDays.com project
Lower Camel Case	A way of writing words with no spaces between them where the first letter of the first word is lowercase and the first letter of each word after is capitalized
MySQL	An open-source relational database management system
Notepad++	A free source code editor
PayPal	A business allowing for payments and money transfers
Producteev	A free web-based task management software [9]
Repository	Basically, a grouped storage location for files and data
Script	A small non-compiled program
Server	A system that responds to requests across a computer network
Set Diagram	A diagram that shows all possible logical relations between a finite collection of sets
Software Requirements Specification Document	A complete description of the behavior of a system to be developed
Source Code	Any collection of computer instructions written in any human-readable programming language

Sponsor	The business the software will be for. The sponsor will provide wants and needs for the software as well as give approval at the end of each stage
Track day event	A day spent at an automotive raceway.
Unified Modeling Language Model	A general-purpose modeling language used to provided sets of graphic notation techniques to create visual models software systems
Upper Camel Case	A way of writing words with no spaces between them where the first letter of the first word is capitalized as well as the first letter of each word after
Use Case	A list of steps, typically defining interactions between a role and a system

Table 1.3.1 - Definitions

ACRONYM	DEFINITION
CSUS	California State University, Sacramento
HTML	HyperText Markup Language
IDE	Integrated Development Environment
PHP	Hypertext Preprocessor
SDS	Software Design Specification
SPMP	Software Project Management Plan
SRS	Software Requirements Specification
SQL	Structured Query Language
STR	Software Test Report
STS	Software Testing Specification
UM	User Manual
UML	Unified Modeling Language
WBS	Work Breakdown Schedule

Table 1.3.2 - Acronyms

ABBREVIATION	DEFINITION
Inc.	Incorporated
Sr.	Senior

Table 1.3.3 - Abbreviations

1.4. REFERENCES

1. Juan Calderon. "www.GotTrackDays.com." *GotTrackDays.* n.p., 2/26/2014. Web. 2/26/2014. <http://www.GotTrackDays.com/>.
2. Juan Calderon. "GotTrackDays, Inc – www.GotTrackDays.com." *Business Plan*. (2014): n.pag. Document.
3. R. Buckley. "Guide to Preparing the PROJECT CHARTER DOCUMENT." (1.30.2014): n.pag. Document.
4. "GitHub." GitHub Inc., n.d. Web. 2/28/2014. <https://github.com/>.
5. "Google+." Google, n.d. Web. 2/28/2014. <www.google.com/+/>.
6. "Google Drive." Google, n.d. Web. 2/28/2014. <http://www.google.com/drive>.
7. "Google Hangouts." Google, n.d. Web. 2/28/2014. <www.google.com/hangouts/>.
8. "Google Wallet." Google, n.d. Web. 2/28/2014. <www.google.com/wallet/>.
9. "Producteev." Producteev, n.d. Web. 2/28/2014. <www.producteev.com/>.

1.5. OVERVIEW OF CONTENTS OF DOCUMENT

This subsection briefly describes each of the remaining sections in the document as well as the contents of each appendix.

2. Product Overview

This section contains a brief description of the project and the deliverables. In addition, a detailed explanation is provided as to how the team intends to manage the project.

3. Project Organization

The project organization section includes an explanation of the phases of work that will be scheduled, measured, monitored, and managed throughout the project's development life cycle. Also included is a description of the team's planned organizational structure for both CSc 190 and CSc 191. Individual team member assignments and responsibilities will be also described.

4. Project Management and Control

This section will address team jBehaving's plans regarding how to keep the project managed and on schedule. Included in this section is a detailed listing of activities and documents used to measure progress and schedule the project's tasks. It also discusses guidelines for dealing with future changes and issues that may arise over the course of the project.

5. Technical Process

This section includes a description of the methods, tools, and techniques the team will use in representing the technical details that will need to be recorded during project development as well as during the development itself. In addition, the team's documentation plan is described along with a list of all documents to be produced over the development life cycle. It will also have some of the coding standards that will be used and how the technical work will be reviewed and approved.

6. Activities and Schedule

This section contains a description of the activities and tasks to be performed in each of the development phases, the resources required to accomplish the work, an estimated (and hypothetical) budget, and the baseline schedule for the project.

7. Document Approvals Page

This section includes signatures indicating the approval of and agreement to the management process, described herein, to be used over the course of the development life cycle.

Appendices

This section contains a copy of the Issue Report Form and Change Request Form, which are management tools used by jBehaving.

2. PROJECT OVERVIEW

This section contains a brief description of the project and the deliverables. In addition, a detailed explanation is provided as to how the team intends to manage the project.

2.1. PROJECT SUMMARY

This project shall require the design and implementation of a website front-end, a back-end database used to store and retrieve data and information, as well as an interface between the two. Track day events are secured ahead of time by Juan Calderon, sole proprietor of Got Track Days, Inc., and he will be selling tickets to these events through the front-end website that jBehaving will be creating [1]. The back-end will consist of a database which will store the event and user information, while the interface will allow the website and database to exchange data.

The software will have many features, which will be presented and described in greater detail in the Software Requirements Specification Document (SRS). The Software Design Specification (SDS) will play a big part in shaping the website and database for this project. It will include many visual aids as well as design techniques to be implemented. After creating an initial version of the software, testing will be planned using the Software Testing Specification Document (STS). Testing will then commence, and a Software Test Report (STR) will be compiled. This report will be used to identify bugs and aid in making the software better than its earlier versions. The software product and all project deliverables will be handed over to the sponsor upon completion of the project, which is at the end of CSc 191.

2.2. PROJECT DELIVERABLES

The project deliverables will include, at a minimum, the Project Charter, the Software Project Management Plan (SPMP), the SRS, the SDS, the STS, and the STR.

During CSc 190, the Charter, the SPMP, and the SRS will be submitted to the team's advisor for grading and approval. The sponsor will also need to approve all of those documents, except for the SPMP, over the course of the first semester. At the end of CSc 190, a final requirements meeting will be held with the sponsor and jBehaving to go over the SRS in detail to assure that all requirements have been accounted for.

The second semester requires that the SDS, STS, and STR also be approved by the faculty advisor. Sponsor approval of these deliverables is not required. At the conclusion of CSc 191, an installation and demonstration meeting will be held with the sponsor. During that meeting, the completed software, a user's manual, copies of all the software development documentation, and the project logs will be delivered to the sponsor on a CD or similar media storage device. Additional items may be identified by the sponsor and/or the team's adviser, and if so, those items will also be delivered to the sponsor at the end of CSc 191.

2.3. EVOLUTION OF THE SPMP

At the start of CSc 190, the work is concentrated on establishing a working relationship with the sponsor, understanding the overall project, developing the team's management plan, and the processes to be used for weekly reviews as well as updates of the work required.

Progress for this project will be thoroughly reviewed during the team's weekly meeting. It is ultimately up to the current team manager to ensure that all members are meeting their individual task deadlines and to make adjustments when necessary. Individual members are required to ask for help if they risk not meeting a deadline and are also responsible for reporting on their task progression at the team meetings.

The Software Project Management Plan will be reviewed, and possibly revised, at the start and end of each phase of work to determine if baseline dates need to be modified. A new Work Breakdown Schedule (WBS) outlining the tasks required by each deliverable will be added to Section 6.1: Activities and Tasks, at the start of each new phase of work. Lastly, over the course of this project, it is possible that the team's management plan may change. If so, those changes will be added to this document.

3. PROJECT ORGANIZATION

The project organization section includes an explanation of the phases of work that will be scheduled, measured, monitored, and managed throughout the project's development life cycle. Also included is a description of the team's planned organizational structure for both CSc 190 and CSc 191. Individual team member assignments and responsibilities will be also described.

3.1. PROCESS MODEL

Documentation:
During the Documentation Phase, the team will develop a Project Charter and the SPMP. The Charter will ensure that the project sponsor, the project advisor and the project team all fully understand each other's points of views about the project at hand. The SPMP will provide a record, guidelines, and practices for how the software development process will be managed and accomplished.

Requirements:
The team will gather information from the sponsor in order to specifically address, understand, and spell out every aspect of the task at hand, and put that information in the SRS. This document will serve as the blueprint for the second semester of the project. The sponsor's involvement with the team, or team representatives, will likely be heaviest during this part of the process.

Design:
During this stage in the software development process, the team will use the SRS to come up with diagrams and specifications for software architecture, operation, and overall system design. Plans for meeting resource and performance goals will be detailed. Requirements traceability will be addressed. This information will be documented in the SDS.

Implementation:
The team will use the SRS and the SDS to begin creating the system. A working software product will be produced. New code and bug fixes will likely be necessary until the testing phase has been completed, so this stage will be ongoing until the system meets acceptance criteria.

Testing:
The Software Testing Specification (STS) document will be prepared to guide the testing of the system. Any bugs or issues identified will be documented. The bugs will then be addressed, code revisions will take place, and the software will be retested. This process will repeat until the software meets acceptance criteria. The STR will be created to show that the product meets requirements as specified in the SRS.

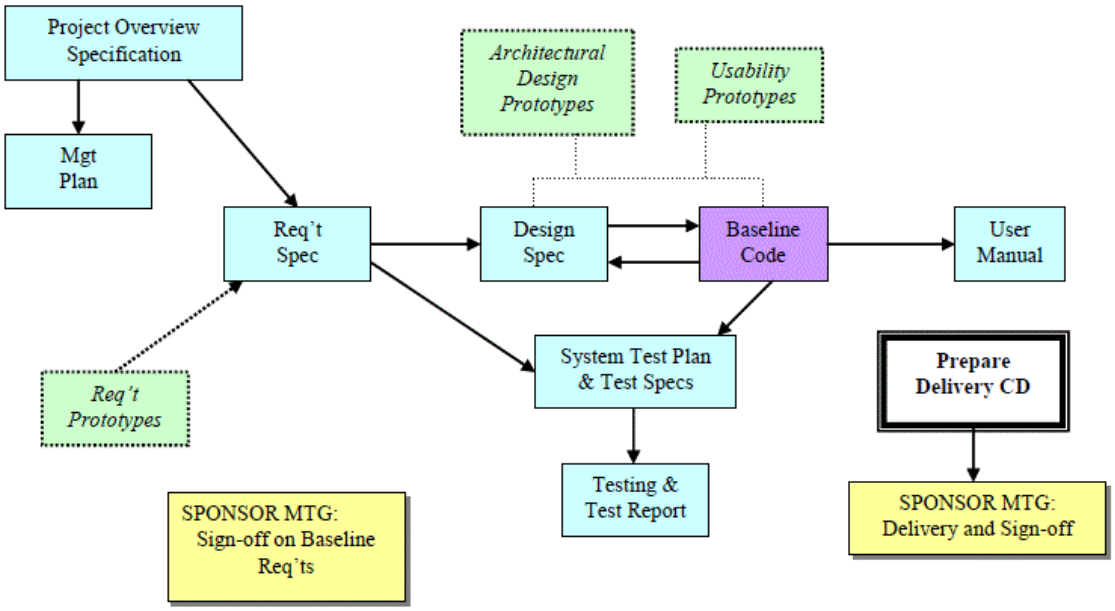

Figure 3.1 – Process Model

3.2. ORGANIZATIONAL STRUCTURE AND INTERFACES

This section details the organization of the team, as well as roles and responsibilities for project manager, team members, faculty advisor, and project sponsor. It also includes a breakdown of responsible parties for each phase of the project.

3.2.1. PROJECT MANAGER

The project manager is the leader of the team and is key to ensuring the success of the development process. The role has several specific duties:

- Delegation of tasks amongst members.
- Updating backlog of tasks.
- Specifying deadline dates (negotiable within team).
- Proofreading final documents before submission.
- Assuring the completion of documents by intended deadlines.
- Making sure that time spent working as a group is logged in the timecard

Because of the academic nature of the project, each team member will be assigned as project manager for a specific time period in order to gain experience in this role. Table 3.2.2 gives the schedule for the rotation of project manager duties.

TIME FRAME	NAME
January 27 through March 10	Cody Lanier
March 11 through April 14	Ashley Finger
April 15 through May 23	Bai Xiong
September 2 through October 13	Michel Watson
October 14 through November 17	Cody Prior
November 18 through December 19	Daniel Gallegos

Table 3.2.1 - Project Manager Rotation Schedule

3.2.2. TEAM MEMBERS

In addition to tasks in developing project deliverables, all team members will have roles in ensuring the proper management of the project. Table 3.2.2 gives each member's assignments. As each team member takes their turn as the project manager the agenda organizer role will pass to that person. The current and previous project manager will exchange other roles as necessary to balance workload.

NAME	PRIMARY	BACKUP
Ashley Finger	Agendas Organizer	Task Creator
Bai Xiong	Stenographer Minutes Organizer	Documentalist Sponsor Liaison Deliverables Printer
Cody Lanier	Documentalist Deliverables Printer Sponsor Liaison	WBS Keeper
Cody Prior	WBS Keeper	Time Keeper
Daniel Gallegos	Task Creator	Decision Tracker Stenographer
Michel Watson	Time Keeper Decision Tracker	Agenda Organizer

3.2.3. FACULTY ADVISOR

The faculty advisor guides the team through the development process, and represents CSUS in monitoring the quality of the work produced. The advisor has several duties:

- Ensure team is making progress on project.
- Provide feedback on the team's work
- Approval of each deliverable before presenting to sponsor.

3.2.4. PROJECT SPONSOR

The role of the project sponsor is to give feedback to the team with regards to his needs and wants, and to give final approval of project deliverables as being accurate and adequate. The sponsor assumes no responsibility for the management of the project.

3.3. PROJECT RESPONSIBILITIES

The current project manager will have primary management responsibility for the phase in progress during their tenure. Their responsibilities with regards to the successful completion of each phase are as follows:

- Assign tasks with deadlines to keep phase on schedule (with input from the team)
- Ensure tasks are being completed on time and adjust schedule as necessary
- Lead technical reviews of deliverables
- Final proofreading of deliverable documents before submission for approval
- Ensuring timely submittal of phase deliverables

Table 3.3 gives a breakdown of the leaders and the main deliverables for each phase:

PHASE	DELIVERABLES	RESPONSIBLE TEAM MEMBERS
Documentation	Project Charter SPMP	Cody Lanier Ashley Finger
Requirements	SRS Feature Wireframes Final Requirements Meeting	Ashley Finger Bai Xiong
Design	SDS	Michel Watson
Implementation	Baseline Code	Cody Prior
Testing	STS STR	Cody Prior Daniel Gallegos
Acceptance	Delivery Meeting	Daniel Gallegos

Table 3.3 – Phases, Deliverables, and Leaders

4. PROJECT MANAGEMENT AND CONTROL

This section will address team jBehaving's plans regarding how to keep the project managed and on schedule. Included in this section is a detailed listing of activities and documents used to measure progress and schedule the project's tasks. It also discusses guidelines for dealing with future changes and issues that may arise over the course of the project.

4.1. PROJECT MANAGEMENT OBJECTIVES AND PRIORITIES.

The senior project has two primary goals. The first is to deliver high quality software that fulfills the sponsor's need. It requires jBehaving to elicit a comprehensive list of accurate requirements and maintain a rigorous schedule of deliverables. The second goal is to prove that jBehaving has satisfied both the goals of the sponsor and the senior project course through extensive documentation of the team's work. In order to meet these challenges within two semesters, several tools will be used to manage the project.

4.1.1. MEETINGS

Team meetings are weekly events that occur each Wednesday from 5:30pm to 6:30pm. Each meeting follows a well-documented agenda that is prepared by the Team Lead in advance. While the contents of each agenda differ, they all follow the same basic structure. The meeting begins by reviewing the minutes of the previous meeting as well as all the previously assigned tasks. The outcomes of the previous meeting are discussed and new tasks are assigned to the team. The rest of the meeting is used for the team to address new and old business items.

jBehaving also has weekly meetings with our advisor, Dr. Meiliu Lu, each Wednesday. The goal of the advisor meeting is to present our progress on the next deliverable and get feedback that is incorporated into the next revision. The sponsor will also give feedback on our baseline schedule and push us to move quicker should we fall behind.

The team will also be meeting with the sponsor, Juan Calderon, throughout the project. Meetings occur whenever the sponsor can find the time to meet (in person or online). These meetings are our primary means of eliciting/verifying requirements.

Records of all three meeting types are kept and saved within the Project Log.

4.1.2. PROJECT LOG

The Project Log is a comprehensive showcase of all the documentation used by jBehaving to manage the project. It is designed to demonstrate that the team has developed the consistency and discipline to manage, schedule, and work on the project from start to finish. The Project Log also serves as an "audit trail" for decisions made by the team during development.

The Project Log is comprised of the following documents:

- ***Meeting Agendas***: The agendas of the numerous team, faculty, and sponsor meetings. These serve to structure the discussion and goals of each meeting.
- ***Meeting Minutes***: A record of what was discussed, brought up, and decided at each meeting.
- ***Timesheets***: Weekly reports on the team's activity towards each deliverable.
- ***Work Breakdown Structure***: Weekly reports on the team's upcoming tasks for a given two week interval.
- ***Baseline Project Schedule/Changes***: The team's initial estimates as to when the main phases of work will begin and complete. The schedule will change and its accuracy will improve over the course of development.
- ***Technical Review Summary Reports***: A record of the team's activity at each deliverable document's technical review.
- ***Decision Traceability Matrix***: A cross-reference table linking key decisions to the date or dates of agendas and minutes in which these key decisions were reported.

4.1.3. OTHER TOOLS:

Producteev[9] is a task management tool that is being used by the team to assign individual tasks and update the team on their progress over time. Producteev is not meant to be a replacement for any contents of the Project Log, but rather an additional tool for keeping progress up to date.

4.2. ASSUMPTIONS, DEPENDENCIES, AND CONSTRAINTS

Due to the nature of the project and the team, there are certain assumptions that can be made before defining conditions of the work to be performed. The assumptions are as follows:

1. jBehaving is a group of students and the project is related to a course requirement at California State University, Sacramento. As such, the timeframe of the project is limited to the Spring 2014 and Fall 2014 academic semesters.
2. As this project is part of coursework for a two semester-unit course, there are expectations on the amount of weekly hours provided by a single student, and these are significantly less than would be expected from a full-time employee.
3. The sponsor understands that the scope of any deliverables will be affected primarily by points one and two.
4. The sponsor's input is necessary to the development process. The sponsor understands that they are expected to provide time for meetings, technical reviews, testing, and approvals as necessary.
5. jBehaving understands that the sponsor is not able to devote themselves full time to the development process and must take this into consideration when soliciting the his time.
6. Any software produced by jBehaving should conform to any applicable standards.
7. The sponsor bears no financial responsibility for the development process.
8. Team jBehaving, to reduce development costs and maintenance costs for the sponsor, will use open source for any software libraries or needed infrastructure when possible.
9. The sponsor may need to provide jBehaving access to infrastructure, such as web hosting services, that are needed for the final delivery of the software
10. The sponsor will maintain full access to any infrastructure or hosting services needed for final delivery.

4.3. RISK MANAGEMENT

The following is a list of possible risks the team may encounter. Included are jBehaving's plans to avoid and mitigate each risk. This list will be updated as new risks reveal themselves over development.

Risk: Assigned Tasks may take longer than original projections.
The primary causes of this risk include mismanagement of personal time or the team's expectations simply not being in-line with the actual difficulty of the task.

Probability: High
Our schedule depends on the timely completion of assigned tasks; delays will have an impact on both our workload and potentially the final deliverable.

Plans to avoid:

- Plan for delays: The schedule should account for the fact that the project will constantly be revised rather than finished in one pass.
- Manage your time: Individual team members should plan out their schedules well in advance and aim to finish early in order to keep the project progressing smoothly. The team lead should also make sure that tasks are being completed on time and to project standards.
- Ask for help: If a team member believes they are unable to finish a task in time, they should seek the help of the team.

Mitigation:
The team lead must quickly reassess the task and the resources need to solve it.

Risk: Requirements change during development
While the team will solicit a large, comprehensive list of requirements, jBehaving recognizes that the needs of the client may still change even as the design phase of the software progresses. jBehaving might also uncover requirements that both the team and sponsor did not originally see before working on the software.

Probability: High
Accurate requirements are key to delivering a quality product.

Plans to avoid:
As previously mentioned, some requirements will not be known until the design process begins. Those hidden requirements can be discovered through further "requirement engineering".

Mitigation:

- If the changes are within the project scope, a change request form will be created and the baseline schedule will be adjusted as needed.
- If the new-found requirements prove to be unreasonable, in terms of the project's scope, the team will attempt to renegotiate with the sponsor to find an alternative.

Risk: Project requires unforeseen technical knowledge
While jBehaving initially expects our past experience with web-based programming to satisfy the website's needs, some requirements may force team members to learn new skills.

Probability: Medium

Plans to avoid:

During the requirements phase, try to research similar projects

Mitigation:

Reassign tasks to members who have the skills and schedule time for learning.

4.4. CHANGE MANAGEMENT

The following is the process that will be taken when changes to the project are being requested or proposed to the team by a team member.

1. A change request will be proposed to the team. Ideally this would take place at the weekly team meeting, but as time is a large factor in this project some changes cannot wait for that long. Alternatively, the change should be proposed either within the jBehaving hangout chat, as a comment within the document itself, or via an email sent to the parties affected by the proposed change.
2. Once a change has been proposed and brought to the team's attention, a formal change request form must be filled out and submitted to the team via the jBehaving Google Drive. The team should be alerted to the official submission of the report by assigning a Producteev task for reviewing the report. In addition, if the change was brought up at a team meeting, a record of the request must be added to the team minutes.
3. Following the submission of a change report, teammates are expected to find some time to discuss (at meetings or online) and review a proposed change. This review is meant to help the team analyze the impact the change would have on the project.
4. Based on the team's assessment of the change impact:
    a. If the change(s) are minor and will not increase the scope of the project, the team will make an immediate decision to grant the change(s) per the request. The change(s) will be communicated to the team, sponsor, and advisor to verify that everyone understands the change(s).
    b. If changes will increase the scope of project, the following will be done:
        i. Present the results of analyzing the impact of the changes to the current progress to the sponsor. The analysis results will include cost of the changes, how severe the changes will be to the current progress of the project, and other additional changes that may occur if the requested changes are granted.
        ii. Negotiate features and functions of project that has highest priority to the sponsor. Other features and functions may be removed to accommodate for the change(s) being granted (if the change(s) are granted) to provide a product of most value to stakeholders.
5. If a change was successfully made, the change is required to be documented. A resolution will be added to the change request form. The form will then be archived for the Project Log.

NOTE: A blank change request form can be found in Appendix B: Change Request Form.

4.5. SCHEDULE CONTROL

Progress towards the objectives of the baseline schedule will be determined by the completion status of tasks. At the baseline start of a deliverable, the team will analyze the requirements of the deliverable and then split the deliverable into a set of small goals. jBehaving will also give a rough estimate of a baseline finish date based on their analysis. At the weekly team meeting, members will be assigned tasks based on the following: Their skills, schedule, and hours contributed to the project. These tasks are officially documented in the weekly team minutes and corresponding Producteev tasks are created. At each successive team meeting, members will report on their individual progress towards the completion of tasks. New tasks will then be assigned accordingly. Based on feedback from the meeting, the team may either reassign tasks to meet the deadline or reschedule the baseline finish date.

NOTE: The estimated schedule can be found in section 6.2: Schedule.

4.6. ISSUE RESOLUTION

Given the scale of the project, issues are bound to arise. If jBehaving is unable to solve the issue immediately, an issue form will be filed by the party that brought up the issue. When a solution to the issue is agreed upon, the form will be updated with the resolution.

NOTE: A blank issue form reporting form can be found in Appendix A: Issue Report Form.

5. TECHNICAL PROCESS

This section includes a description of the methods, tools, and techniques the team will use in representing the technical details that will need to be recorded during project development as well as during the development itself. In addition, the team's documentation plan is described along with a list of all documents to be produced over the development life cycle. It will also have some of the coding standards that will be used and how the technical work will be reviewed and approved.

5.1 METHODS, TOOLS, AND TECHNIQUES

During the development life cycle of this project, the jBehaving team will be using several methods, tools, and techniques to help document the technical details of the GotTrackDays.com project. There will also be several coding standards that the team will use to help keep the software as easy to read as possible and so there is accountability for code written.

In the SRS, the team will be using the UML model (a visual mapping of the architecture of the system) to represent various aspects of the software system to be developed.

- Use Cases will be used to show the flow of work a user can encounter while using the website.
- Set diagrams will be used to show when and how much certain items will overlap in data.
- Entity Relationship Diagrams will be used to depict the entire system. Each class will be represented as well as how they relate to other classes within the system.

Tools to be used:

- GitHub[4] will be used as a way to collaboratively share work between team members through cloud storage of their repository
- Google Apps will be integrated into the final website to allow for a better user interface
- Google+ will be used as a communication source between team members
- Google Hangouts[7] will be used as a primary communication source between team members and also the sponsor
- Google Drive[6] will be used to store and work on documents collaboratively by the entire team
- Google Wallet[8] will be integrated into the final website to allow customers to pay for events
- PayPal will be integrated into the final website to allow customers to pay for events
- Producteev[9] will be used by the entire team to help keep track of assigned tasks and their completion
- Notepad++ or a similar text editor will be used as the primary source code editor (the team may choose to add in an IDE or other editor at a later time)

Programming Languages to be used:

- MySQL
- PHP
- HTML
- JavaScript

Coding Conventions to be used:

- Each member will have a comment before the method/function written saying who coded it and what its main purpose is
- Class names will use upper camel case
- Method/function names will use lower camel case
- Variable names will use lower camel case
- Each team member will abide by the "if it isn't broken, don't fix it" mentality before editing code
- If a team member edits code they will leave a comment saying which parts they added or removed and the reason for doing so
- Hard coded values will be avoided whenever possible to ensure the highest level of portability
- Testing will be implemented on the software before committing it to the repository to avoid having to search through thousands of lines of code for an error

5.2. SOFTWARE DOCUMENTATION

Throughout the development of the software, team jBehaving will be in contact with each other very often. The main form of communication will be conducted through Google Hangouts and weekly team meetings. These communications enable a better understanding of what is expected of each team member and allows simple questions to be asked as well as answered.

At each team meeting, each member will be assigned specific tasks that correlate with the current document that is being prepared. These assignments will then be posted in the team's Producteev account and assigned to the proper member. Once the task is completed, the assignee will need to check off, in Producteev, that the task is complete and then, create a new assignment for their partner to review their work. The peer review will help to minimize the time that the team needs to spend doing the technical review before submitting the document as well as provide an additional proof read.

At the team meeting following the peer reviews, the technical review for the document will be completed by the whole team. During the technical review, the team will review each section one at a time and note any changes or concerns they have. Once final revisions have been completed, the team will sign the approval page signifying that each team member fully accepts the document as it stands. This will complete the first draft of the technical document and it will then be submitted to the team's advisor.

Once the advisor has read through the document and explained to jBehaving any corrections that need to be made, one person will be assigned the task of making the revisions. The assignee will need to bring it to the following meeting for a team review. The document will be re-submitted to the advisor after the team review. This process will repeat until there are no more changes needing to be made, according to the advisor. The advisor will then sign the approval page signifying the completion of the document.

After the document has been accepted as complete by the advisor, the team will present the document to the sponsor to read through. Once jBehaving has completed the document, the team

members, sponsor and advisor will sign the approval page. This final approval will establish the document as the baseline version.

5.3. DOCUMENTS

Spring 2014

Project Charter *02/19/2014 - 03/12/2014* *Team Lead: Cody Lanier*

The purpose of the Project Charter is to define the reasons for undertaking the project along with any objectives and constraints. Within this document there will be directions concerning the project, a high level risk management plan, a communication plan, in-scope and out-of-scope items, as well as the identity of the proprietor of Got Track Days Inc., Juan Calderon.

SPMP *03/05/2014 - 04/02/2014* *Team Lead: Ashley Finger*

The SPMP document will summarize the project and outline the plan the team has for all the deliverables of the project. It will contain a project, risk and change management plan, the technical process, a schedule of when the deliverables will be finished and an activities/task plan to get them all accomplished.

Feature Wireframes *03/19/2014 - 04/30/2014* *Team Lead: Ashley Finger and Bai Xiong*

The Feature Wireframes will serve as a mockup of what the actual user interface will look like once the project is complete. It will contain snapshots of what the website will look like at different stages of a given use case. The most important key features will have wireframes created.

SRS *03/19/2014 - 04/30/2014* *Team Lead: Ashley Finger and Bai Xiong*

The SRS document enlists all the necessary requirements that are required for the project development. It will discuss the purpose of the project along with a system overview of what the project will look like in the end. It will also contain very detailed requirements for the system, user, hardware, software, and communication interfaces as well as the functionality, performance and design specifications. There will be information on the reliability, availability, security, maintainability, and portability of the software to be developed as well as detailed information on the database system that will be used. This document will contain all the information needed to understand the project in great detail and as a whole, and all the information necessary to determine all success criteria for the project.

Fall 2014

Architectural Design *09/03/2014 - 09/17/2014* *Team Lead: Michel Watson*

The Architectural Design will show the different parts of the software system and how they relate to one another. This document will serve as a basis of how to create the software so that the proper parts of it can interact and others cannot. It will also give a very good estimate of exactly how big the project is, and how many different pieces will need to be brought together to make the whole project work as desired.

Feature Wireframes *09/10/2014 - 09/24/2014* *Team Lead: Michel Watson*

The Feature Wireframes will serve as a mockup of what the actual user interface will look like once the project is complete. It will contain snapshots of what the website will look like at different stages of a given use case. The most important key features will have wireframes created. This document will differ from the previous Feature Wireframes document because the team will be more involved in the design process at this point and therefore have a better understanding of exactly what the final product will look like.

SDS *09/10/2014 - 10/01/2014* *Team Lead: Michel Watson and Cody Prior*

The SDS document will contain explicit information as to how the product is to be put together as well as all necessary drawings and diagrams to help illustrate this procedure. Within this document there will be material about pattern usage, algorithms to be used, how the interface will work, as well as the interaction of the system and its many components. This document will serve as the basis for how the code will be written so that it meets the standards of the project.

Baseline Code *10/01/2014 - 11/12/2014* *Team Lead: Cody Prior*

The Baseline Code will be the source code of the project. The code will be well documented throughout the software development lifecycle so that it is easy for the sponsor to add to and/or debug anything after the course of this project. The source code will follow the design and requirements set forth in the SDS and SRS very closely.

STS *11/12/2014 - 11/19/2014* *Team Lead: Cody Prior*

The STS document will be written to provide full details as to how the software will be tested. It will enumerate out specific examples of which features will need to be tested and how the testing will occur. This document will also contain standards that team will agree upon which state what needs to happen in order for the software to be considered as passing the tests and what would be considered failing.

STR *11/19/2014 - 12/03/2014* *Team Lead: Cody Prior and Daniel Gallegos*

In the STR document, the team will specify which tests passed and which ones failed. If a test failed, a reason for the failure will be stated as well. The team will then need to clearly state how the code will be changed to fix the failing test and make plans to retest the software once the code is fixed. Each test will examine a part of the software that has been listed as a requirement in the SRS and therefore must pass in order for the software to be considered fully functional.

UM *12/03/2014 - 12/10/2014* *Team Lead: Daniel Gallegos*
The user manual (UM) will be the final stage of the GotTrackDays.com project. Once everything is running as desired, the team will construct a user manual with information on how to properly use the website. The user manual will have all the basic how-to's so that our sponsor knows how to operate the website successfully. Also, a few troubleshooting tips will be included in case he runs into an issue.

Team Lead:
The Team Lead will have the responsibility of leading team and advisor meetings each week, as well as any sponsor meetings that happen during his/her term. He/She will be in charge of making the agendas for each meeting, delegating tasks to all team members, keeping team members on track, and resolving any impediments during the development process. Each team lead has been given a specific start and end date. Therefore, each member will have the opportunity to be the team lead.

The team lead schedule is as follows:

TIME FRAME	NAME
January 27 through March 10	Cody Lanier
March 11 through April 14	Ashley Finger
April 15 through May 23	Bai Xiong
September 2 through October 13	Michel Watson
October 14 through November 17	Cody Prior
November 18 through December 19	Daniel Gallegos

Table 5.3 - Project Manager Rotation Schedule

6. ACTIVITIES AND SCHEDULE

This section contains a description of the activities and tasks to be performed in each of the development phases, the resources required to accomplish the work, an estimated (and hypothetical) budget, and the baseline schedule for the project.

6.1. ACTIVITIES AND TASKS

The first phase, Software Requirements Specification, of this project development is shown below with the specific tasks and break down of the schedule during that phase. The WBS includes the specific baseline dates for starting and completing the various tasks for the phase. Once a subsequent phase has begun, a change to the Software Project Management Plan will allow for the inclusion of the WBS.

Upon completion of each activity, a written document will be produced. In addition, updating the WBS will occur during team management meetings. Additional training will be added to the schedule as needed. More in depth descriptions such as the types of training needed, team members involved, pre/post-entry criteria, and the training method will be included when training phases occur.

Tasks	Baseline Duration	Baseline Start	Baseline Finish
Phase: SRS	6 weeks	03/19/2014	04/30/2014
Gather Requirements	2 week	03/19/2014	04/02/2014
Featured WireFrames	1 week	04/02/2014	04/09/2014
Use Cases	1 week	04/09/2014	04/16/2014
Use Case Modeling	1 week	04/16/2014	04/23/2014
Technical Review	1 week	04/23/2014	04/30/2014

Table 6.1 - SRS WBS

The training plan will serve as the guide for the team when learning needs to be completed of a new tool or concept. Training sessions will be held at the start of a phase to ensure successful completion of that section. This plan includes the methods to use, team members involved, and criteria entering and exiting the training. The template is as follows:

1. Members participating will be chosen based on the workload and tasks that have been assigned. Any other interested members can also take part as well.
2. Set a goal for the training event. Each session has a set goal that will be achieved. In other words, this is the reason why the training is being scheduled.
3. Set entry criteria that specify the current knowledge the members of the training participants have of the topic. This serves as a base for the training session.
4. Training Methods:
    a. Reading - Documentation of programming languages and concepts are widely available online and in hard copies. Hard copy textbooks are also available online and in libraries.
    b. Tutorials - This training method provides hands-on learning experience for more effective learning. Knowledge about the topic will be applied to example scenarios to learn how to use the new information.
5. Set exiting criteria that specifies the expected knowledge obtained once the training session is complete. If the criteria are achieved, the training session is deemed a success; otherwise another training session will be scheduled.

6.2. SCHEDULE

The baseline schedule is established to track the team's progress through the various development phases. Provided below is a table that specifies the dates agreed upon by the team for each deliverable. In addition, the Gantt chart below shows a visual representation of the schedule dates for each deliverable. The schedule will be approved as a baseline of the project. Additional changes that occur throughout the project development will be made to the schedule as required.

NOTE: Refer to section 5.3 of this document for further detail of each deliverable.

Deliverable	Start Date	End Date
Project Charter	02/19/2014	03/19/2014
Software Project Management Plan	03/05/2014	04/02/2014
Featured Wireframes	03/19/2014	04/30/2014
Software Requirements Specification	03/19/2014	04/30/2014
Architectural Design	09/03/2014	09/17/2014
Featured Wireframes	09/10/2014	09/24/2014
Software Design Specification	09/10/2014	10/01/2014
Baseline Code	10/1/2014	11/12/2014
Software Testing Specification	11/12/2014	11/19/2014
Software Testing Report	11/19/2014	12/03/2014
User Manual	12/03/2014	12/10/2014

Table 6.2.1 - Baseline Schedule

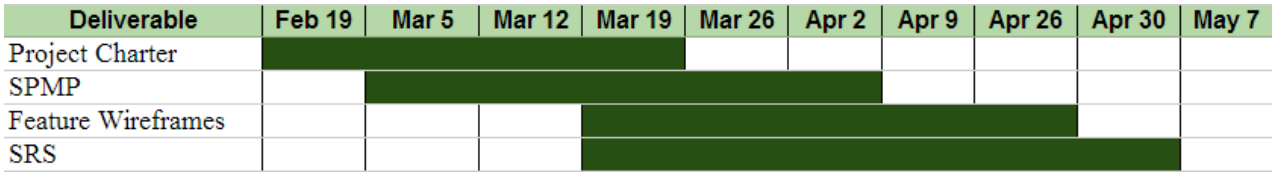

Table 6.2.2 - Gantt Chart Spring 2014

Deliverable	Sept 3	Sept 10	Sept 17	Sept 24	Oct 1	Oct 15	Oct 29	Nov 12	Nov 19	Dec 3	Dec 10
Architectural Design											
Feature Wireframes											
SDS											
Baseline Code											
STS											
STR											
User Manual											

Table 6.2.3 - Gantt Chart Fall 2014

6.3. RESOURCE REQUIREMENTS

Specific resources will be needed throughout the course of the project. jBehaving is not expected to provide those resources at the expense of the team. Necessary resources will be paid for by the sponsor and therefore, a budget is not required. Descriptions of how resources will be acquired are provided in the following list.

Materials

- Printing - Ink will be bought from office supplies stores.
  - Campus printing is provided throughout campus via printers. Paper and ink for printing are provided by CSUS.

Software & Tools

- Google Drive[6] - provided by Google as an free open source tool
- Google Hangout[7] - provided by Google as a free open source tool for communication. Also available to all users with a Gmail account.
- Google Apps - provided as free open source software
- GitHub[4] - provided to all users as a free open source software for version control of development projects
- Producteev[9] - provided to all users as a free task tracking tool via the web
- Eclipse or similar IDE - open source tools available to all users

Time (16 weeks for 190 and 16 weeks for 191)

- 1 hour 15 minutes of class time each week
- 1 hour team meeting per week (additional meetings may be necessary)
- 30 minute advisor meeting per week
- 30 minutes - 1 hour Sponsor meeting, minimum 2 per month
- 1-4 additional hours per week, per team member

Other

- Professional Graphics Designer - Sponsor will be the point of contact with the graphics designer. Graphics produced by the designer will be provided to the team through the Sponsor.

7. DOCUMENT APPROVAL PAGE

This section includes signatures indicating the approval of and agreement to the management process to be used over the course of the development life cycle.

By signing below, I hereby agree to and approve of the proposed Project Management Plan, to be used over the course of the development life cycle of the project.

Advisor:

Dr. Meiliu Lu: _____ **Date:**_____

jBehaving Development Team:

Ashley Finger: _____ **Date:**_____

Bai Xiong: _____ **Date:**_____

Cody Lanier: _____ **Date:**_____

Cody Prior: _____ **Date:**_____

Daniel Gallegos: _____ **Date:**_____

Michel Watson: _____ **Date:**_____

APPENDIX A: ISSUE REPORT FORM

jBehaving Issue Report Form

* Required

Issue Reported By *

Which category does the issue pertain to?

- Management
- Charter
- Requirements
- Wireframes
- Design
- Testing

Describe the issue with as much detail as possible *

Submit

Never submit passwords through Google Forms.

APPENDIX B: CHANGE REQUEST FORM

jBehaving Change Request Form

* Required

Change Requested By *

Which category does the change request pertain to? *

- Management
- Charter
- Requirements
- Wireframes
- Design
- Testing

Describe your change request with as much detail as possible. *

Why is this change needed? *

In your opinion, will this change affect any delivery dates? *

- Yes
- No
- Maybe

Submit

Never submit passwords through Google Forms.