Project Plan

Project Glacier

Revision History

Revision	Date	Comments
0.1	9/6/16	Initial Draft
0.2	9/27/16	Updated project synopsis

Table Of Contents

Document Overview

This document provides processes and procedures that the management and development team should follow. The procedures listed in this document are used to manage and monitor the team by team leaders, thus providing clear guidelines for responsibilities of team members.

Project Overview

Glimpse.io is a tool that provides users the ability to connect to various data sources, such as a spreadsheet or relational database and create data visualizations. The application relies upon existing open-source data visualization tools vega and vega-lite developed by the University of Washington's Interactive Data Lab. Unlike many other data visualization tools, Glimpse does not constrain the user to select from pre-existing templates and encourages experimentation with various ways to interactively explore data.

Project Scope

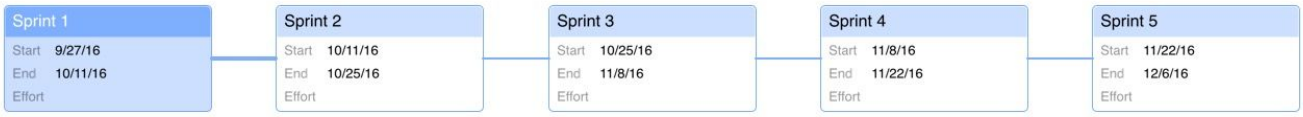
The project will be completed by May 18, 2017 and consist of developing Glean and Glance libraries. These libraries will fall under one 'umbrella' project called Glacier, that can be utilized by data visualization applications. These libraries will be written in Typescript to handle modeling and data configuration. The model layer and corresponding API is the primary deliverable for the project and will include support for caching data, storing database and other data source credentials, and data sources used in the visualization and formatting for the visualization itself. To support this the client has given us permission to develop supporting libraries and hosting runtimes. These supporting libraries may include, javascript data access libraries, visualization generation libraries, and a javascript frontend library.For an end user this means what we expect to have done would be a javascript-based reference implementation that exercises the model and API provided by Glacier. The final deliverable will not include a full user interface however one may be created for testing purposes. Development of a complete native GUI is not within the scope of this project. It is anticipated that the Glacier library will run in a variety of different Javascript engines with various execution contexts, the deliverable is only required to run and pass tests on a single implementation of the V8 Javascript engine used by Node.js.

Deliverables

- Glean and Glance Libraries
- Electron application
    - The Electron application is not required by the sponsor however to thoroughly test our project developing an application which consumes these libraries would best demonstrate the working product.

Schedule and estimates

Sprint schedule for the first semester



Team Roles and Responsibilities

The purpose of these guidelines is to provide clarity to the roles and responsibilities of various team members.

Team leaders are be responsible for meeting agendas and coordinating meetings with the sponsor.

Meeting Scribe is responsible for taking any necessary notes during meetings. This will be necessary when we collect and prioritize requirements from the sponsor.

The **Scrum Master** will be responsible for running daily standup meetings, backlog grooming, leading sprint retrospectives, and sprint planning meetings.

Developers will be responsible for reporting status at daily standups.

	Role
Marc Prud'hommeaux	Project owner
Bill Stumbo	Sponsor Coach
Colin Ferris	Team Leader, Developer
Wesley Wigham	Developer
Steven Greenberg	Developer
Zach Calfin	Developer, Meeting Scribe

Development Process

The project will be implemented utilizing Scrum, and using tools such as Xcode, GitHub, ZenHub,Visio, and Microsoft Project.

Developers are required to submit a pull request before code is merged into the master branch. This will help maintain a clean repository and provide transparency to the sponsor and team. The sponsor will be notified via notification when a developer submits a pull request. Developers may merge their code into master branch once they've received feedback from at least one other teammate.

Risk Management

Initial Project Risks

Risk	Description	Mitigation Strategy
Scope Creep	Scope creep is a major concern when using an Agile methodology. Changes can affect the delivery time of tasks.	We will implement a change management process.
Vega / Vega Lite	Vega / Vega Lite are libraries used by our application. Since these products aren't owned by us there's risk using software developed by someone else.	Monitor libraries for any unexpected changes. If any updates break existing project code, attempt to fix the project as quickly as possible. If any pull requests are slowing down project progress attempt to get them merged as quickly as

		possible, otherwise find a work around.
Beta Software	Use of unstable beta versions of software causes unexpected consequences when software doesn't work as expected	If beta software acts in an unsuspecting manner, attempt to solve the problem without switching out software. If that is not possible, switch back to the latest stable version of software that is causing the issue
Slow Pull request acceptance	Slow pull requests on open source software changes	Attempt to have the pull request reviewed as quickly as possible. If the pull request can not be reviewed in a timely manner, fork the branch and create a custom package for use in our project
External Dependencies	The project has a lot of external dependencies that are not controlled by the team which may cause slowdowns when the dependencies are modified	Monitor all dependencies for changes on a regular basis and make sure all dependencies are performing as expected
Unfamiliar technologies	The use of new technologies slows down initial performance due to the lack of familiarity among team members	Move the more mission critical tasks to developers who have a better understanding of the technologies until the rest of the team becomes more comfortable with new technologies
Illness	Widespread illness throughout RIT causes team members to become sick and not perform to their full capacity	Rest as much as possible and take steps to speed up recovery. Soup, meds and vitamins. Take steps to prevent catching any additional illness such as hand washing and sanitization. Get proper amounts of rest and eat a

		well balanced diet that provides the proper nutrients for supporting the immune system
Midterms	Midterm exams at RIT require team members focus causing less time to be spent working on the project	Attempt to manage time to allow for exam preparation as well as time to be spent working on the project
Team member jobs	Team members jobs occupy a significant amount of free time, taking away from their available time to work on the project	Attempt to manage time to allow for working on both the project and work that needs to be completed for jobs. If possible, try to commit to less hours at work to spend more time on the project. Create a time management plan that will allow for hours to be spent both on work and the project. Spend as much free time as possible working on the project.

Measurements & Metrics

Velocity - velocity will be calculated at the conclusion of a sprint. Velocity will help us measure how much work can be completed in a sprint.
Developer happiness - This metric will be also used to monitor the team and mitigate any problems. Slight changes in this number can indicate things are going really well or really bad. Happiness will be measured on an individual level at standups on a scale from 1(worst) to 5 (best).

Communication and reporting

Each team member is responsible for posting weekly status reports that include time / effort worked and tasks completed that week.