**Software Requirements Specification**

**for**

**Sphinx**

**Version 3.0**

**Prepared by Team Venture**

**October 10th, 2013**

**Table of Contents**

[**1.0 Purpose**](#h.3h531q27g1hh)

[**2.0 Assumptions**](#h.azpshbmvv7sv)

[**2.1 Scope**](#h.nix40dwlzpp)

**2.2 Data**

**2.3 Definition of Terms**

**2.3.1 Project management entities**

**2.3.2 Project management events**

**2.3.3 Sphinx Terms**

[**3.0 Functional Requirements**](#h.w8v10ozblu1g)

**3.1 Authentication/Authorization**

**3.2 Information Feeds**

**3.3 Entity Interactivity**

**3.4 Entity and Catalog Search**

**3.5 Entity Subscription**

**3.6 System Administration**

**3.7 System Administration Wish-list**

[**4.0 Non-Functional Requirements**](#h.m54pd4kj9ptm)

**4.1 Usability**

**4.2 Security**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| **SRS** | **9/24/2013** | **Initial Revision** | **V 1.0** |
| **SRS** | **10/3/13** | **Sponsor Feedback** | **V 2.0** |
| **SRS** | **10/10/13** | **Sponsor Feedback** | **V 3.0** |

# 1.0 Purpose

A web based project management subscription system that interfaces with different project management system’s APIs to allow users of the project management system to subscribe to and comment on projects being worked on within a company.

The initial implementation of this system will interact with an API for the JIRA project management system, as requested. Once the implementation for JIRA is complete, the ability to interface with more project management systems will be completed.

Two Sigma is a technology company that applies a rigorous, scientific method-based approach to investment management. The goal of this project is to provide visibility and communication about projects within an organization. Employees will be able to browse through their companies projects using various search criteria, subscribe to projects, and get periodic updates about their subscribed projects, when aspects about them change. Users of the system will also be able to comment on projects with their feedback and vote if they like or dislike a project.

The system will be developed over the next thirty weeks by a team of four. The team will work closely with their faculty coach and sponsor to assure that deadlines are met on time and every three weeks, a prototype of the system will be ready for testing.

# 2.0 Assumptions

## 2.1 Scope

### 2.1.1 In Scope

The Sphinx system will consist of the following components:



|  |  |
| --- | --- |
| **High Level System Component** | **Description** |
| Project update ingester | The project update ingester will process project updates and ideas and persist them in the Sphinx database. |
| Project subscription manager | The project subscription manager will capture project subscription details for employees and persist them in the Sphinx database. |
| Project update publisher | The project update publisher will apply employee-specific subscription filters to project updates and ideas and deliver the filtered updates to the employee. |
| Idea feedback gatherer | The idea feedback gatherer captures votes and feedback on the ideas from employees and passes them on to the project management system. |
| Web UI | The Web UI will … |

### 2.1.2 Out of Scope

1. Create customized JIRA API for Two Sigma; This will be created and provided by Two Sigma, however the API spec will be created by team Venture.
2. Accounts that will be used to subscribe to projects: Single sign on
3. Any type of email notifications.

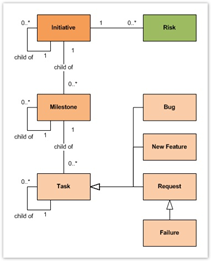
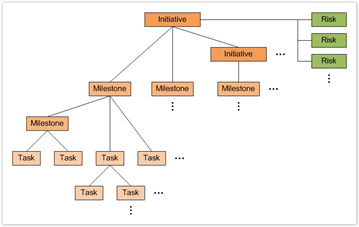
**2.2 Data**

Data for the Sphinx system will be sourced from a set of APIs provided by Two Sigma. Please see Data/Service Specification.

**2.3 Definition of Terms**

**2.3.1 Project Management(PM) Entities**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Initiative** | An initiative is the minimum viable set of deliverables to achieve a defined goal. An initiative groups work items together through milestones. Has a well-defined scope, has a start and end date, is sponsored by one person (business owner). One person is held accountable to deliver the initiative (assignee). An initiative is a unit of prioritization for the business owner. |
| **Milestone** | A milestone is an event that receives special attention. It is part of the initiative reporting and can group work items. One person is held accountable to deliver the milestone (assignee). |
| **Risk** | A project risk is an uncertain event that, if it occurs, has a positive or negative effect on the prospects of achieving project objectives. |



# 

**2.3.2. Project Management (PM) Events (from External PM System)**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Change Event | A Change Event is a collection of field changes for when a Project Management Entity is created, updated, or deleted. |
| Report Event | Report Events describe the submission of a report about an Initiative, Milestone, or Risk. Generally, the reports will provide a description of what happened to an Initiative, Milestone, or Risk over a period of time. Some Report types include weekly status, weekly plan, monthly accomplished , and monthly upcoming.  While Change events may happen at any time, Report events will generally be generated at the end of a week (for the weekly reports) or end of a month for monthly reports. They should be treated in a similar fashion as Change Events should be (e.g. users can view them, subscribe to them) |

**2.3.2. Sphinx Terms**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Query Subscription | A subscription to the result of a frequently run saved query. |
| Entity Subscription | A subscription to all of the Sphinx Events associated with a particular PM Entity |
| Sphinx Event | Sphinx events would be derivative events that Sphinx would be emitting based on configurable rules. For instance, the system may receive a Change event describing that the “Due Date” of an Initiative has changed. Based on the configured rules, Sphinx may determine that the event should also be classified as a “Schedule Event” because a date field has changed. We wanted to push the Sphinx Events to the system because it would be easier for us to tweak the configurable rules as needed (e.g. create a new kind of event, consolidate events)  Sphinx Events are additional classifications of events that will be received from Jira (you can think of it as the system adding an additional “sphinx type” field to the events that it receives). Users should be able to view, search for, and subscribe to sphinx events as other events in the system. |

3.0 Functional Requirements

The main functional areas of the system include:

|  |  |
| --- | --- |
| **Area** | **Description** |
| Authentication / Authorization | Users will be able to log in with basic credentials with room for future ability to sign in using SSO |
| Information Feeds | Information feeds shall be received from an external API and displayed to the user. |
| Entity Interactivity | Entities (such as milestones, risks, and initiatives) will be interactive so that a user can perform actions such as commenting and voting on them. |
| Entity Catalog and Search | Searching and filtering the catalog of entities. |
| Entity Subscription | Creating saves queries and custom subscriptions. |
| System Administration | Modifying the Sphinx system settings. |

3.1 Authentication / Authorization:

* The system shall provide a standalone interface for users to login with username/password credentials
* The system shall provide a configurable option to disable the username/password login in favor of a servlet filter –based single-sign-on solution (SSO)
* System shall be able to assign the proper roles to each user once authenticated (e.g. Administrator role)

3.2 Information Feeds:

* User’s homepage shall initially have a feed of their current projects
  + A user’s initiatives / milestones might explicitly name them as “assignee”, “reporter”, “watchers” or a “businessOwner”
* User’s homepage shall initially have a feed of new projects across the firm
* User’s homepage shall initially have a feed of team projects coming up
  + A user’s team’s initiatives would be those where the “businessGroup” or “providerGroup” is set to the user’s team (which would be defined within the “groups” field in the “user” object)
* The user’s homepage shall check for updates every X minutes
  + This *should* be configurable, with a default of 15 mins
* Subscribed upon queries should be able to be viewed in the user’s personal feed

3.3 Entity Interactivity:

* The user shall be able to click on a “Learn More” link on the displayed initiative to be taken to the entity’s profile page.
  + There will be a different page for Initiatives, Milestones, and Risks.
    - Can all look very similar
  + Each initiative profile page will provide more detailed information and it will hold all comments and votes on the entity in question.
* User will be able to comment on entities. Those comments will pass through to the PM System.
* User shall be able to vote on entities. Those comments will pass through to the PM System.
* User should be able to vote/watch an entity from the dashboard
  + Voting sends a watch request back to JIRA
* User shall be able to comment on an initiative from the dashboard page

3.4 Entity Catalog and Search:

* The user shall be able to go to a ‘catalog’ page and initially see a listing of all projects they have access to
* The user shall be able to sort the catalog/search results alphabetically, by date of most recent change or by date proposed
* The user shall be able to search projects by keywords, by department or by a search query of their choice
* The user shall be able to perform a faceted search for entities and/or events of interest
* The user shall be able to save a query from the search results screen as a subscription for future use

3.5 Entity Subscription:

* User shall be able to choose subscription queries by picking from a checklist of facets such as the sponsor team, sponsor, and initiative’s business unit.
* User shall be shown a screen allowing them to pick events that they view, a name for their subscription query, and a description for the subscription query.
* User shall be able to view all subscriptions and subscription queries on the subscriptions page
* User shall be able to edit subscription queries from the subscription screen
* User shall be able to subscribe to entities. Subscribed entities will be added to a user’s subscriptions.
* User shall be able to subscribe to an initiative from the dashboard

3.6 System Administration

* Administrators shall be able to change different settings regarding its connection to Jira
  + Admin shall be able to change how often JIRA events are pulled
  + Admin shall be able to change the URL of service endpoints
* Administrators shall be able to assign other users as administrators
* Administrators shall be able to define and modify refined Sphinx Events based on a set of configurable rules though the use of an XML file or something similar
* Administrators will be able to manually synchronize Sphinx with the source PM system.

3.7 System Administration Wish-list

*Items that may be out of scope that will be brought into scope if time at the end of the project allows.*

* Administrators shall be able to view the subscriptions of the user community.
* Administrations shall be able to view current system metrics to ensure that the system is stable
  + How many events have been processed in the past day, week, month
  + How many users have set custom subscriptions in the past day, week, month
  + How many total events have we processed in the system
  + How many users have voted / watched / commented through the system

# 4.0 Non-Functional Requirements

## 4.1 Usability Requirements

* UI will use a form of bootstrap
* UI will use industry standard approach
* Events will stream to the user on the dashboard
* The app will be single page
* App will run on mainly Firefox and chrome

## 4.2 Security Requirements

* The system should use a servlet based filter for authenticating users that can be swapped out with Two Sigma’s internal authentication filter
* The system shall filter out entities / events that the user does not have permission to access
  + Each initiative, milestone and risk has an “allowedAccessUsers” field that defines the set of user that are permissioned to view that entity. If that field is blank, then all the system users can view that entity.