**Requirements Engineering Report**

**Date: February 17, 2016**

**Version v1.00**

**Software Project Name**

Prescribe

**By:** Team StarMony

***Project Manager:*** Jeremy Brown

***Quality Assurance:*** Brandyn Deffinbaugh

***Technical Lead:*** Mitchell Powell

**Table of Contents**

1. Introduction
   1. Objective
   2. Scope
   3. Success Criteria
   4. Collaboration with Stakeholders
2. Project Plan
   1. WBS
   2. Project Resources
   3. Responsibility Matrix
   4. Gantt Chart
   5. Pert Chart
   6. Cost Estimation
      1. Function Point Estimation
      2. LOC Estimation
      3. Cost Estimations
   7. Risk Plan
   8. Project Monitoring & Control Mechanisms
3. Requirements & Analysis Models
   1. Major Software Functions
   2. Use Case Diagrams
   3. Use Case Descriptions
   4. Activity Diagrams
   5. Sequence Diagrams
   6. Requirements Class Models
   7. Prototype Description
   8. Data Directory
   9. Limitations & Constraints
   10. Non-functional Requirements
4. Problems Encountered
5. Bibliography
6. Introduction
   1. Project Objective

The objective of our project is to provide a web application that can aid users in the discovery of new music based on their musical preferences.

* 1. Project Scope

The user will be able to input an artist or a band into search field which will provide a similar list of artists or bands. If the user creates an account using Facebook or Google+, they will be able to give feedback to train our model by up or down voting the search results. This will help grow the machine to give better feedback to our users. The user can also post their search results to their social media page. Search results will provide the user with a list of similar bands including the bands’ information such as their biography and discography. The user also be able to favorite a band for later reference. We also want to include a top artists section that users can view to see which artists have been given the most up votes for the week, month, or year.

* 1. Success Criteria
     1. Search Functionality
     2. Rating system integration
     3. Smooth Web GUI
     4. Icon graphics
  2. Collaboration with Stakeholders

• End-Users

• Facebook

o Mark Zuckerberg

• Google

• Spotify

• Pandora

1. Project Plan
   1. Work Breakdown Structure



* 1. Project Resources

|  |  |  |  |
| --- | --- | --- | --- |
| Resource Name | Cost | Description | Status |
| Eclipse IDE | Free | Internal Development Environment for Java programming language | Obtained |
| Domain Name | $20 / year | The domain name at which the web application will be hosted | In Progress |
| JavaScript Interpreter/Web Browser | Free | Interpreter for the JavaScript programming language, as the front end will be developed in JavaScript | Obtained |
| Labor | $9375 | 1.5 person/months at a rate of $6,250 per person month | In Progress |
| Web Hosting Server Space | $10 / month | Space on a server to host the data needs of the Prescribe project. | In Progress |

Total: $9375 up front, $140 recurring annually (this will certainly need to be updated after a little more analysis about what will be required for the project)

* 1. Responsibility Matrix



* 1. Gantt Chart



* 1. Pert Chart



* 1. Cost Estimation
     1. Function Point Estimation
     2. Lines of Code Estimation
     3. Cost Estimation
  2. Risk Plan
  3. Project Monitoring & Control Mechanisms

There are several mechanisms that the Starmony group will utilize in order to ensure effective communication and proper maintenance of the code base. As a group, we will meet a minimum of once a week in a formal, face-to-face meeting to discuss the progress of the project, any issues that have arisen and for code review. In addition, the group will maintain regular group-wide electronic communication to ensure that all group members are kept up to date with what progress has been made, short term goals, and current issues. The group will maintain a “TODO” document on the Git repository to ensure that there is effective communication about what current issues are at hand.

It will be the responsibility of all group members to pull any changes from the remote repository whenever they work on their local branches so any integration issues can be caught early and fixed before they become a larger problem. In turn, it will also be the responsibility of each group member to push any completed changes or additions to the master branch of the remote repository as often as possible. Any works in progress should be maintained in a development branch to ensure a compiling codebase on the master branch at all times.

1. Requirements & Analysis Models
   1. Major Software Functions
   2. Use Case Diagrams
   3. Use Case Descriptions
   4. Activity Diagrams
   5. Sequence Diagrams
   6. Requirements Class Models
   7. Prototype Description
   8. Data Directory
   9. Limitations & Constraints
   10. Non-functional Requirements
2. Problems Encountered
3. Bibliography