Software Requirement Specification Document

FlyTranscription

Version 1.0

Prepared by Basheer Becerra

Illinois State University

April 25th, 2016

Table of Contents

1. Revision History 3
2. Introduction 4
2.1 Purpose 4
2.2 Product Scope 4
3. Overall Description 4
3.1 Product Perspective 4
3.2 Product Functions 4
3.3 User Classes and Characteristics 4
3.4 Operating Environment 5
3.5 Design and Implementation Constraints 5
3.6 User Documentation 5
4. External Interface Requirements 5
4.1 User Interfaces 5
4.2 Software Interfaces 5
4.3 Communications Interfaces 6
5. System Features 6
5.1 Gene Input 6
5.1.1 Status – Required 6
5.1.2 Description 6
5.2 Result Output 6
5.2.1 Status – Required 6
5.2.2 Description 6
5.3 Result Score 6
5.3.1 Status – Required 6
5.3.2 Description 6
5.4 Result Visualization 6
5.4.1 Status – Required 6
5.4.2 Description 6
5.5 Result Export 7
5.5.1 Status – Optional 7
5.5.2 Description 7
6. Nonfunctional Requirements 7

6.1 Security Requirements 7
Appendix A: Glossary 7
Appendix B: Analysis Model 8
B.1 Software Architecture Design 8
B.2 Query Process Flow 8

1. Revision History

DATE	DESCRIPTION	AUTHOR	COMMENTS
05/01/2016	Initial requirements documentation	Basheer Becerra	Initial Documentation

2. Introduction

2.1 Purpose

The purpose of the document is to allow the developers and stakeholders to introduce, describe, and provide detailed analysis about FlyTranscription. For technical developers, the document will provide in-depth knowledge on the various technical aspects of the project such as architecture, data sources, external libraries and APIs, etc. For application users, it provides a description of functionalities that the application contains. For stakeholders, it provides functionality and the scope of the application.

2.2 Product Scope

FlyTranscription is a web-based application tool that takes an inputted gene within the *Drosophila* and *Anopheles* genome and returns all enriched transcription factors that are shared among other genes with similar patterns. Data for the genome and transcription factors are retrieved through the Intermine web-service client, which reads data from the Flybase database.

3. Overall Description

3.1 Product Perspective

FlyTranscription is unique as it allows the ability to transcription factors among a list of similar *Drosophila* and *Anopheles* genes from the input of a single gene. For instance, a test may be used to determine the reason a *Drosophila* gene is not being encoded to express an immune response to a venom. A predicted gene may be inputted to determine enriched transcription factors to test and determine the cause of an autoimmunity response in flies. The list of enriched transcription factors can then be tested by the scientist to complete the objective of the experiment. The value received from this process reduces the amount of work and assumptions in finding a potential list of transcription factors manually.

3.2 Product Functions

The user will enter a gene name. The application will then search for other genes with similar pattern (through factors such as tissue, development stage, similar enhancers, etc.) and generate a list of genes. With the list of genes, the application will use ChIP sequencing to determine the binding sites of the gene and ultimately receiving a list of enriched transcription factors that are shared among the list of genes. Both lists of transcription factors and genes will be outputted with a score to measure the accuracy of each result. The algorithm of the scoring system is to be developed throughout the process of implementation. Also, the algorithm for find the list of genes with similar patterns is to be developed throughout the process of implementation.

3.3 User Classes and Characteristics

The users of the application are primarily directed to the average biologist at universities working with flies and fly genetics. Users can range from a large setting such as a research intensive lab to a small setting such as classrooms for educational purposes.

3.4 Operating Environment

Since FlyTranscription is web-based, the tool can be accessed on all modern operating systems and browsers with Internet access. For biological visualization to work appropriately, the browser must have JavaScript enabled.

3.5 Design and Implementation Constraints

The main foreseen constraint of the application is if Intermine, FlyTranscription's data source, doesn't produce analyzable data results such as producing too many or too little sets within a given query/filter. The data retrieved from Intermine must have the right kind of information and resolution for the application.

3.6 User Documentation

Optionally, an online user manual and a video tutorial will be available on the website for application users.

4. External Interface Requirements

4.1 User Interfaces

FlyTranscription will not be dependent on any third party user interfaces for its functionality.

4.2 Software Interfaces

FlyTranscription will be dependent on the following third party software interfaces for its functionality:

- Intermine – An open source Java web-service client to execute queries to the integrated FlyMine data source.
  - http://intermine.org/
- BioJava – An open source Java framework to process biological data.
  - http://biojava.org/
- BioJS – An open source JavaScript library to visualize biological data.
  - http://www.biojs.net
- Bootstrap – An open source HTML, CSS, and JavaScript framework for responsive website development.
  - http://getbootstrap.com/
- jQuery – A JavaScript library containing many features such as HTML document traversal, manipulation, etc.
  - http://jquery.com/
- Datatables – A table plugin for the jQuery library that includes pagination, filtration, etc.
  - https://datatables.net/
- Spring Framework – An open source application framework which provides core features for web-based application such as security (Spring Security) and RESTful web-service communication.
  - https://spring.io/

4.3 Communications Interfaces

For the requirements, FlyTranscription will not be dependent on any third party communications interfaces for its functionality. However, if FlyTranscription is offering user login and authentication, HTTPS communication may be necessary.

5. System Features

5.1 Gene Input

5.1.1 Status – Required

5.1.2 Description

The user must be able to input a single gene name within a form (following typical gene nomenclature and conventions). If the gene is not available, a validation error should be displayed on the screen. Validation should be checked on the front-end (by JavaScript and AJAX) and on the back-end.

5.2 Result Output

5.2.1 Status – Required

5.2.2 Description

The results will contain a list of genes that were used for the analysis (which are genes with similar expression patterns compared to the inputted gene). Each gene will also contain supplement information such as the gene secondary identifier, gene symbol, chromosome DB identifier, chromosome location start, chromosome location end, and chromosome location strand. The results will also contain a list of enriched transcription factors based on of the list of genes. Each transcription factor will also contain supplement information such as the data source name, regulatory regions factor, factor symbol, sequence residue, and score (as described in “5.3 Result Score”).

5.3 Result Score

5.3.1 Status – Required

5.3.2 Description

The result for each transcription factor and gene will contain a score measuring the accuracy of the data. The algorithm used to determine the score is yet to be determined.

5.4 Result Visualization

5.4.1 Status – Required

5.4.2 Description

The results will display two tables. The first table will have rows containing the list of similar genes. The second table will have rows containing the list of transcription factors. Each column of both tables will contain supplement information as described in “5.2.2 Description”. Each column header of both tables will contain a tooltip of what the data means/indicates (mainly for educational purposes). Each table data will contain a tooltip with detailed information on the specified data. Each table data will also be a link to its information page on the Flymine website.

5.5 Result Export

5.5.1 Status – Optional

5.5.2 Description

The query result can be exported as an image file (for any biological visualization) or a CSV file (for any tabular list of data). There should be a button displayed on the screen for the user to optionally download the exported file.

6. Nonfunctional Requirements

6.1 Security Requirements

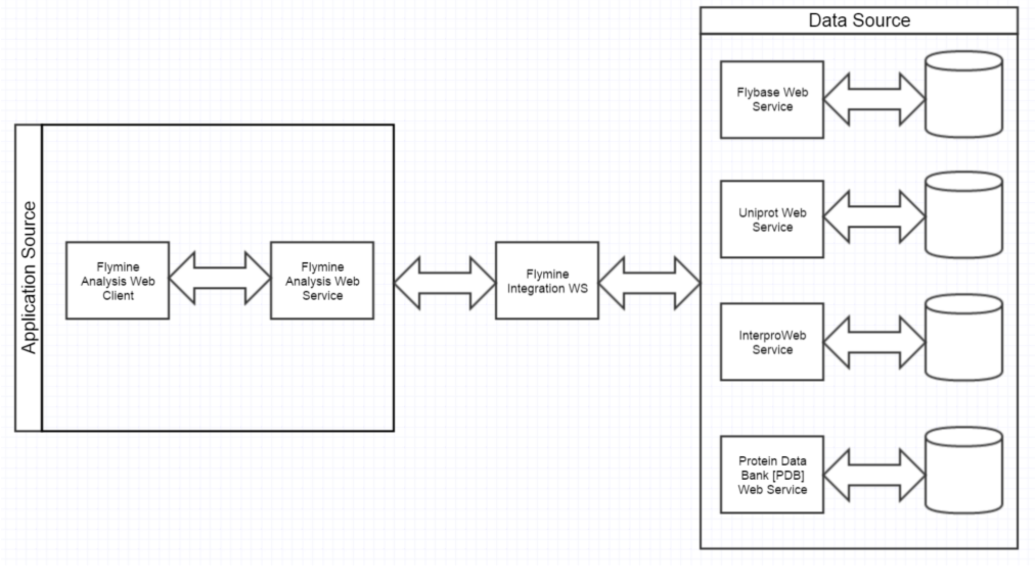
Since all data retrieved from Intermine does not contain any Sensitive Personal Information (SPI) and the tool does not involve any login process or user authentication, there are no security requirement on the protection of any data. Although, secure measures must be taken to prevent common website vulnerabilities such as Cross-Site Request Forgery (CSRF) and Cross-Site Scripting (XSS).

Appendix A: Glossary

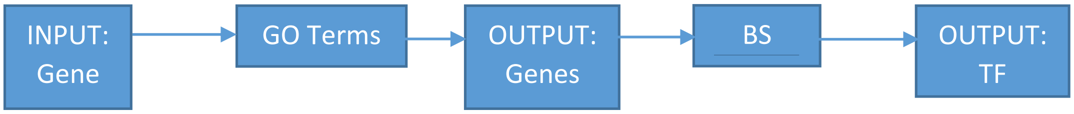
- *Drosophila* – A group of species (or genus) that generally include small flies.
- *Anopheles* – A group of species (or genus) that generally include mosquitos.
- Genome – A set of all DNA within a containing species or group.
- Gene – A particular region of DNA that encodes a functionality.
- Transcription Factor (TF) – A protein that binds to a gene to control the rate at which it is encoded.
- Enrichment – A general method used to enhance, refine, or otherwise improve raw data.
- Expression – The process at which a gene functionality is apparent.
- ChIP sequence – A process to determine the binding sites of a gene, hence receiving the transcription factors for a gene.
- GO – A way to search genes based off of a property.
- AJAX – A technology that provides the ability to send and receive data without refreshing the browser page.
- Front-end – Processes that occur on the client computer (AKA the web browser).
- Back-end – Processes that occur on the server computer.

Appendix B: Analysis Model

B.1 Software Architecture Design



B.2 Query Process Flow


INPUT: Gene
GO Terms
OUTPUT: Genes
BS
OUTPUT: TF