Vision Document

Paul Cain

RNA-Seq Analysis Pipeline

Table of Contents

1.0 Introduction

This vision document will provide an overview of the system to be developed and define in detail the use cases and specific requirements of the system.

1.1 Definition of Terms

- **Base:** A, C, T, or G.
- **BLAST:** A queryable program and database used align a large number of sequences very quickly to see which one match the query sequence the best. For more information see [1].
- **Beocat:** A Beowulf cluster used for supercomputing. For more information see: [2].
- **Cufflinks package:** A set of programs used for reference RNA-seq analysis. For more information see [3].
- **De-novo:** When a processing analysis workflow has no genome to use as a reference to assist in the process of RNA-seq.
- **Differential expression:** When a cell produced more or less of a gene or transcript in under one condition than it does under another condition.
- **Fasta:** A file in fasta format.
- **Fasta format:** A file format used to store DNA, RNA, or protein sequences. For more information see [4].
- **Fastq:** A file in fastq format.
- **Fastq format:** A modified version of fasta format that also stores information about the quality of the reads. For more information see [5].
- **FDR:** False discovery rate. FDR is a statistical correction meant to adjust a p-value to make it more accurate. For more information see [6].
- **FPKM:** A normalized way to compare transcript abundance. For more information see [7].
- **Gene:** A sequence of DNA that defines a specific function in an organism. For more information see [8].
- **Gene Ontology (GO):** A controlled vocabulary used to define and discuss genes. For more information see [9].
- **GTF Format:** "A file format used to hold information about gene structure" [10].
- **Meaningful results:** Subjectively determined by the user of the system.
- **Novel:** Something that is not currently known, such as an undiscovered gene or transcript.
- **Pairwise format:** One of the formats in which BLAST output can be displayed. For more information, see [11].
- **Processing analysis:** The part of the system that handles processing the reads, including quality filtering, transcript assembly, differential expression testing, and other steps related to processing the reads.
- **Processing analysis workflow:** One of the workflows used for processing analysis. The different workflows give different options for how to do processing analysis.
- **P-value:** The likelihood that the tested phenomena happened by chance. For more information see [12].
- **Query analysis:** The part of the system that handles querying either the regular database or the BLAST database so that the user can query for useful information about their data.
- **Read:** A short sequence used for input in processing analysis.
- **Reference:** When a processing analysis workflow uses a pre-existing reference genome to assist in the process of RNA-seq.
- **RNA-seq:** Using high-throughput technologies to discover information about RNA. For more information see [13].
- **RNA-seq Analysis:** The analysis of data produced by RNA-seq. For more information, see [14].
- **Sequence:** A sequence of bases.
- **Tophat:** A program used to map reads to a reference genome, the output of which is given to cufflinks. For more information see [16].

- **Transcript:** Reads are assembled into transcripts during RNA-seq and then various other tests can be performed on them. A gene can be comprised of multiple transcripts.
- **Trimmed fastq file(s):** Fastq files that have had their low quality bases removed.
- **Trinity package:** A set of programs used for de-novo RNA-seq analysis. For more information, see [15].

2.0 Overview

2.1 Overview of the Project

This project, formally as the RNA-Seq Analysis Pipeline, is designed to simplify the process of RNA-seq analysis by providing the ability to filter read input files for quality, do differential expression testing, find novel genes transcripts, and store the results for later querying.

2.2 Purpose

The purpose of the system is to automate and simplify as much as possible the process of RNA-Seq analysis, both in processing the reads and in examining the final output after the reads have been processed.

2.3 Goals

Goal 1: Provide a system through which Biologists can do RNA-seq analysis.
Goal 2: The system should allow the user to get meaningful results with minimal learning time. For this goal to be satisfied, two senior Biologists familiar with RNA-seq must approve the system.

2.4 Risks

- **Risk 1:** The system is not easy enough to use.
    - Description: Because this system is designed to simplify the RNA-seq analysis process by taking many existing RNA-seq analysis tools and glueing them together, hiding the unnecessary details of how it all works, the biggest risk is that the system will not be significantly easier to use that alternatives such as Galaxy.
    - Mitigation: To mitigate this, the system will be designed with simplicity, ease of use, and usability as one of the main goals and feedback will be integrated into the development process. One example of designing for simplicity, ease of use, and usability is that helpful hints and tips will be included on each page.
- **Risk 2:** RNA-seq data is too large to store on the Bioinformatics center's server.
    - Description: Because RNA-seq data can become very large, sometimes exceeding more than a gigabyte in size for the reads files,, the amount of data stored in the system's database could exceed a terabyte or more after prolonged usage.
    - Mitigation: Ensure adequate storage capacity on the Bioinformatic center's servers and have a mechanism to remove data from the database that is no longer needed.
- **Risk 3:** The database is too slow.
    - Description: When databases reach an extremely large size, a long delay between issuing a query and receiving the result can occur.
    - Mitigation: Select a DBMS that can handle large datasets, optimize the database as needed, and/or make the databases distributed.
- **Risk 4:** Processing analysis workflows take too long to run.
    - Description: Processing multiple samples of large amounts of data with the various RNA-seq programs can take a long time.
    - Mitigation: Using concurrency to have multiple samples be processed at the same time and provide a web page that can be bookmarked where the user can go to check the status of their currently running processing analysis workflow.

2.5 Constraints

- The system must run on Linux.
- The system must run on Beocat.
- The system should have some type of web interface.
- The license must allow the system to be free for academic use.

2.6 Features

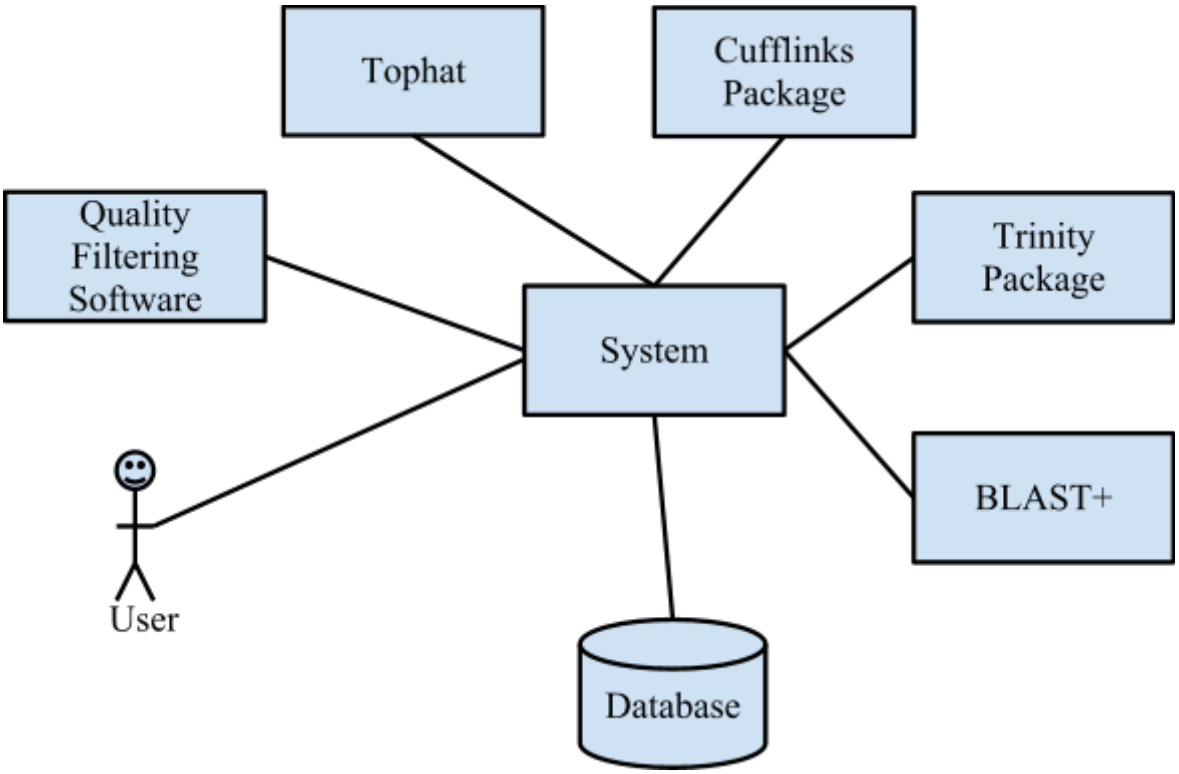
The features of what the system does are described by the use cases in section 3.1 of this document.

2.7 Quality Attributes

- **Database performance:** The risk of having a large database like this is that it may become slow as more and more data is added to it. How best to measure and optimize database performance has yet to be determined.
- **Correctness of results:** If the system gives incorrect results, that could lead to Biologists drawing incorrect conclusions. To confirm correct results, the output of the system will be compared to output manually generated by the programs used by the system, such as Tophat and Cufflinks.
- **Ease of use:** One of the important goals of the system is for it to be easy to use such that not a lot of time and effort is needed to learn the system before it can start producing meaningful results. This quality attribute is qualitatively determined by the Biologists. At least two senior Biologists familiar with RNA-seq must approve the system.
- **Data Integrity:** Because the data stored in the database is important to Biological researchers such that data loss or theft of the data would be a major concern, the system should take measures to both secure the data from unauthorized access and prevent data integrity problems. This quality attribute will be implemented by creating some type of database redundancy and implementing authentication and authorization for the system.
- **Browser compatibility:** The web interface of this system should run on at least Internet Explorer 8 and higher, the latest version of Firefox, and the latest version of Chrome.

2.8 External Interfaces

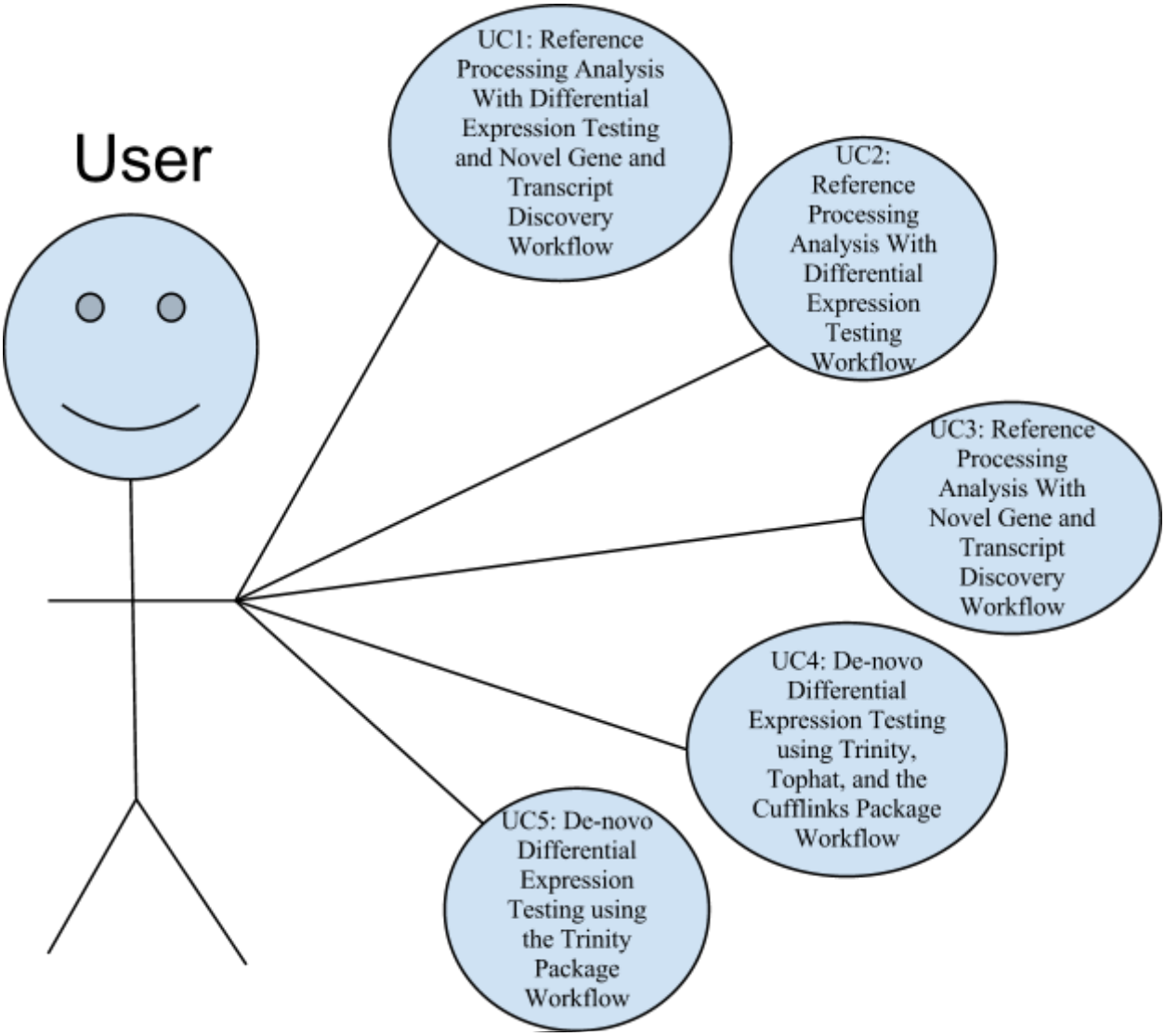
Because the system glues many different RNA-seq programs together, it has many external interfaces.


Tophat
Cufflinks Package
Quality Filtering Software
Trinity Package
System
BLAST+
User
Database

3.0 Requirements Specification

3.1 Use Cases

3.1.1 Processing Analysis Use Cases



3.1.1.1 Use Case 1: Reference Processing Analysis with Differential Expression Testing and Novel Gene and Transcript Discovery Workflow

Name: Reference Processing Analysis with Differential Expression Testing and Novel Gene and Transcript Discovery Workflow
Identifier: UC1
Actors: User

Description: The user uses the system to obtain novel gene and transcript information and differential expression information about an organism for whom they have fastq read files and whose reference genome is possessed by either the user or the system.

Precondition(s):

- The user has fastq read file(s) from at least two different samples.
- The user knows whether their fastq read files are in Sanger, Solexa, or Illumina1.3+ format.
- The user has one fastq read file per sample if using single end reads or two fastq files per sample if using paired end reads.
- The user or the system has a reference genome that the user finds suitable.

Postcondition(s):

- The system has provided the user with novel gene and transcript information about their data.
- The system has provided the user with differential expression information about their data.
- The user has stored the differential expression information and novel genes and transcript information on the system if they desired.

Basic Course of Action:

1. The system prompts the user to enter the required and optional inputs for the workflow.
2. The user enters the required inputs and any options inputs they wish to enter.
3. The system processes the input from the user.
4. The system uses Tophat to map the reads to the reference genome.
5. The system uses Cufflinks to assemble the transcripts.
6. The system uses Cuffcompare to combine the transcripts.
7. The system uses Cuffdiff to do differential expression testing and novel gene and transcript discovery.
8. The system presents the final output files to the user for downloading and/or saving to the database.
9. The user downloads and examines whichever files they want.
10. The user decides to store the results for later querying. [Alternate Course A: The user decides not to store the results in the database for later querying]
11. The system stores the results in the database for later querying.
12. The use case ends.

Alternate Course B: The User Decides Not to Store the Results in the Database for Later Querying

A.10. The use case ends.

3.1.1.2 Use Case 2: Reference Processing Analysis with Differential Expression Testing Workflow

Name: Reference Processing Analysis with Novel Gene and Transcript Discovery Workflow

Identifier: UC2

Actors: User

Description: The user uses the system to obtain differential expression information about an organism for whom they have fastq read files and whose reference genome is possessed by either the user or the system.

Precondition(s):

- The user has fastq read file(s) for at least one sample.
- The user knows whether their fastq read files are in Sanger, Solexa, or Illumina1.3+ format.
- The user has one fastq read file per sample if using single end reads or two fastq files per sample if using paired end reads.
- The user or the system has a reference genome that the user finds suitable.

Postcondition(s):

- The system has provided the user with differential expression information about their data.
- The user has stored the differential expression information on the system if they desired.

Basic Course of Action:

1. The system prompts the user to enter the required and optional inputs for the workflow.
2. The user enters the required inputs and any optional inputs they wish to enter.
3. The system processes the input from the user.
4. The system uses Tophat to map the reads to the reference genome.
5. The system uses Cuffdiff to do differential expression testing.
6. The system presents the final output files to the user for downloading and/or saving to the database.
7. The user downloads and examines whichever files they want.
8. The user decides to store the results for later querying. [Alternate Course A: The user decides not to store the results in the database for later querying]
9. The system stores the results in the database for later querying.
10. The use case ends.

Alternate Course B: The User Decides Not to Store the Results in the Database for Later Querying

A.8. The use case ends.

3.1.1.3 Use Case 3: Reference Processing Analysis with Novel Gene and Transcript Discovery Workflow

Name: Reference Processing Analysis with Novel Gene and Transcript Discovery Workflow
Identifier: UC3
Actors: User
Description: The user uses the system to obtain novel gene and transcript information about an organism for whom they have fastq read files and whose reference genome is possessed by either the user or the system.
Precondition(s):

- The user has fastq read file(s) for at least one sample.
- The user knows whether their fastq read files are in Sanger, Solexa, or Illumina1.3+ format.
- The user has one fastq read file per sample if using single end reads or two fastq files per sample if using paired end reads.
- The user or the system has a reference genome that the user finds suitable.

Postcondition(s):

- The system has provided the user with novel gene and transcript information about their data.
- The user has stored the novel genes and transcript information on the system if they desired.

Basic Course of Action:

1. The system prompts the user to enter the required and optional inputs for the workflow.
2. The user enters the required inputs and any optional inputs they wish to enter.
3. The system processes the input from the user.
4. The system uses Tophat to map the reads to the reference genome.
5. The system uses Cufflinks to assemble the transcripts.
6. The system uses Cuffcompare to combine the transcripts and find the novel genes and transcripts.
7. The system presents the final output files to the user for downloading and/or saving to the database.
8. The user downloads and examines whichever files they want.
9. The user decides to store the results for later querying. [Alternate Course A: The user decides not to store the results in the database for later querying]
10. The system stores the results in the database for later querying.
11. The use case ends.

Alternate Course A: The User Decides Not to Store the Results in the Database for Later Querying

B.9. The use case ends.

3.1.1.4 Use Case 4: De-novo Differential Expression Testing using Trinity, Tophat, and the Cufflinks Package Workflow

Name: De-novo processing analysis for differential expression
Identifier: UC4
Actors: User
Description: The user uses the system to obtain differential expression information about an organism whose fastq reads files they have.
Precondition(s):

- The user has fastq read file(s) for at least two different samples.
- The user knows whether their fastq read files are in Sanger, Solexa, or Illumina1.3+ format.
- The user has one fastq read file per sample if using single end reads or two fastq files per sample if using paired end reads.
- The uses wants to do de-novo differential expression testing, but wants to use Tophat and the Cufflinks package to do it.

Postcondition(s):

- The user has obtained differential expression information about their data.
- The user has stored the differential expression information on the system if they desired to.

Basic Course of Action:

1. The system prompts the user to enter the required and optional inputs for the workflow.
2. The user enters the required inputs and any optional inputs they wish to enter.
3. The system processes the input from the user.
4. The system uses Trinity to do a de-novo assembly of all the reads from all the samples combined.
5. The system creates a Bowtie index of Trinity's output.
6. The system uses Tophat to map the reads to the Bowtie index made from Trinity's output.
7. The system uses Cufflinks to assemble the transcripts.
8. The system uses Cuffcompare to combine the transcripts.
9. The system uses Cuffdiff to do differential expression testing.
10. The system presents the final output files to the user for downloading and/or saving to the database.
11. The user downloads and examines whichever files they want.
12. The user decides to store the results for later querying. [Alternate Course B: The user decides not to store the results in the database for later querying]
13. The system stores the results in the database for later querying.
14. The use case ends.

Alternate Course B: The User Decides Not to Store the Results in the Database for Later Querying

B.12. The use case ends.

3.1.1.5 Use Case 5: De-novo Differential Expression Testing using the Trinity Package Workflow

Name: De-novo processing analysis for differential expression
Identifier: UC5
Actors: User
Description: The user uses the system to obtain differential expression information about an organism whose fastq reads files they have.
Precondition(s):

- The user has fastq read file(s) for at least two different samples.
- The user knows whether their fastq read files are in Sanger, Solexa, or Illumina1.3+ format.

- The user has one fastq read file per sample if using single end reads or two fastq files per sample if using paired end reads.
- The user prefers the Trinity package for de-novo differential expression analysis.

Postcondition(s):

- The user has obtained differential expression information about their data.
- The user has stored the differential expression information on the system if they desired to.

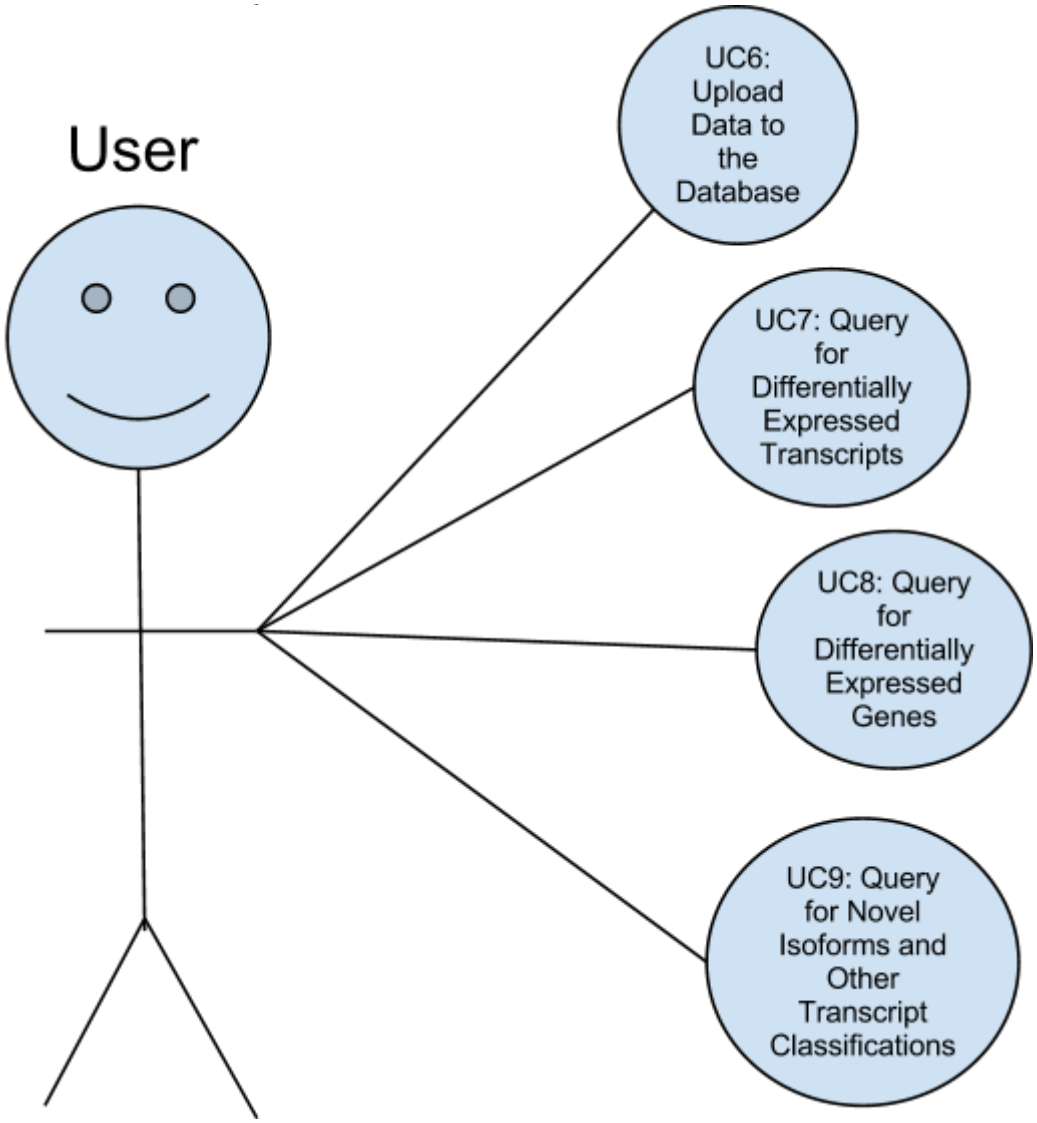
Basic Course of Action:

1. The system prompts the user to enter the required and optional inputs for the workflow.
2. The user enters the required inputs and any optional inputs they wish to enter.
3. The system processes the input from the user.
4. The system uses Trinity to do a de-novo assembly of all the reads from all the samples combined.
5. The system aligns the reads from each of the individual samples against the assembled transcripts produced by Trinity.
6. The system uses RSEM to estimate the abundance of reads.
7. The system uses the merge_RSEM_frag_counts_single_table.pl script to combine the RSEM abundance values from the different samples into a single table.
8. The system runs edgeR to do differential expression testing between the various samples.
9. The system presents the final output files to the user for downloading and/or saving to the database.
10. The user downloads and examines whichever files they want.
11. The user decides to store the results for later querying. [Alternate Course B: The user decides not to store the results in the database for later querying]
12. The system stores the results in the database for later querying.
13. The use case ends.

Alternate Course B: The User Decides Not to Store the Results in the Database for Later Querying

B.11. The use case ends.

3.1.2 Query Analysis Use Cases



3.1.2.1 Use Case 6: Upload Data to the Database

Name: Upload data to database
Identifier: UC6
Actors: User
Description: The user uploads the data required for querying, which is only done once for each data set.
Precondition(s):

- The user has some data on which they would like to run some queries.
- The specific data that the user wants to upload is not already in the database.

Postcondition(s):

- The user's data has been uploaded to the database.

Basic Course of Action:

1. The system prompts the user to enter the name of their data set and select the type of data they have.

2. The user enters a name for their data set, selects the type of data set, and uploads the files related to the data set.
3. The system processes the user input and stores the data in the database.
4. The system displays a message to the user indicating success.
5. The use case ends.

3.1.2.2 Use Case 7: Query for Differentially Expressed Transcripts

Name: Query for differentially expressed transcripts
Identifier: UC7
Actors: User
Description: The user uses the system to do a query on transcript differential expression data.
Precondition(s):

- The user wishes to query a given set of transcript differential expression data using some set of criteria for some purpose.
- The transcript differential expression data that the user wishes to query is stored on the system.

Postcondition(s):

- The query results contain the data filtered and organized in the way specified by the user.
- The query results have been displayed to the user and are available for download.

Basic Course of Action:

1. The system presents the user with the queryable data sets available to them and the available query options.
2. The user selects the data set they wish to query.
3. The user selects an FDR value for their query.
4. The user optionally selects filter settings for GO number, GO name, transcript length, and/or Cuffdiff-assigned transcript name.
5. The user submits the query.
6. The system processes the query.
7. The system displays the query results to the user and makes them available for download.
8. The use case ends.

3.1.2.3 Use Case 8: Query for Differentially Expressed Genes

Name: Query for Differentially Expressed Genes
Identifier: UC8
Actors: User
Description: The user uses the system to do a query on gene differential expression data.
Precondition(s):

- The user wishes to query a given set of gene differential expression data using some set of criteria for some purpose.
- The gene differential expression data that the user wishes to query is stored on the system.

Postcondition(s):

- The query results contain the data filtered and organized in the way specified by the user.
- The query results have been displayed to the user and are available for download.

Basic Course of Action:

1. The system presents the user with the queryable data sets available to them and the available query options.

2. The user selects the data set they wish to query.
3. The user selects an FDR value for their query.
4. The user optionally selects filter settings for GO number, GO name, and/or Cuffdiff-assigned gene name.
5. The user submits the query.
6. The system processes the query.
7. The system displays the query results to the user and makes them available for download.
8. The use case ends.

3.1.2.4 Use Case 9: Query for Novel Isoforms and Other Transcript Classifications

Name: Query for Novel Isoforms and Other Transcript Classifications
Identifier: UC9
Actors: User
Description: The user uses the system to do a query on transcript isoform data.
Precondition(s):

- The user wishes to query transcript isoform data using some set of criteria for some purpose.
- The transcript isoform data that the user wishes to query is stored on the system.

Postcondition(s):

- The query results contain the data filtered and organized in the way specified by the user.
- The query results have been displayed to the user and are available for download.

Basic Course of Action:

1. The system presents the user with the queryable data sets available to them and the available query options.
2. The user selects the data set they wish to query.
3. The user optionally selects filter settings for class code, GO number, GO name, transcript length, and/or Cuffdiff-assigned transcript name.
4. The user submits the query.
5. The system processes the query.
6. The system displays the query results to the user and makes them available for download.
7. The use case ends.

3.1.2.5 Use Case 10: Query the BLASTable Database

Name: Query the BLASTable database
Identifier: UC10
Actors: User
Description: The user uses the system to query a BLAST database.
Precondition(s):

- The user wishes to query a BLASTable database using some criteria for some purpose.
- The sequence data that the user wishes to query is stored on the system in a BLAST database.

Postcondition(s):

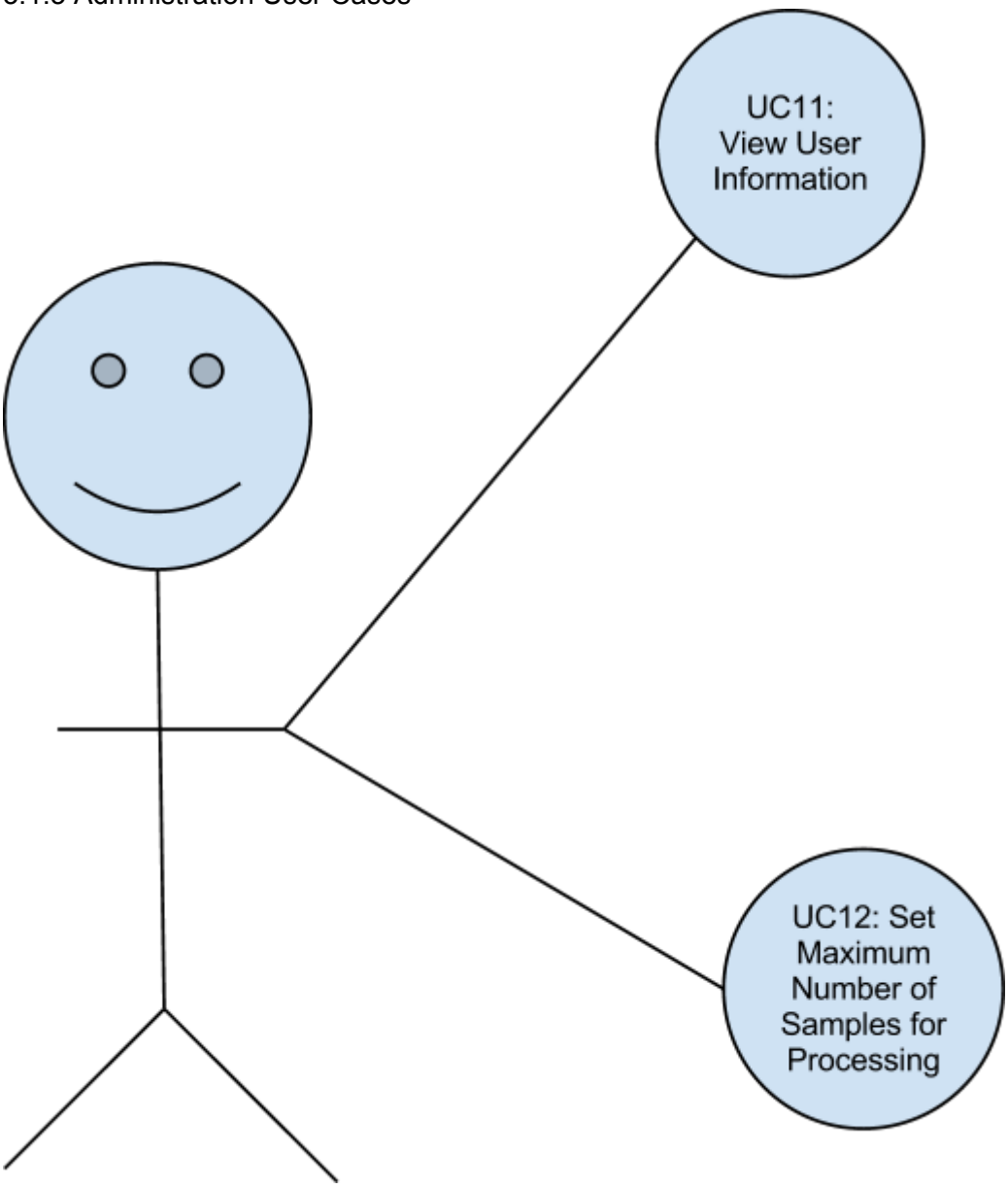
- The BLAST results were created using the exact settings specified by the user.
- The BLAST results have been displayed to the user.

Basic Course of Action:

1. The system presents the user with the BLASTable data sets available to them and the available query options.

2. The user selects the data set they wish to BLAST against.
3. The user inputs the sequences they would like to use to query the BLAST database.
4. The user optionally customizes the BLAST settings.
5. The use submits the BLAST query.
6. The system displays the BLAST results to the user.
7. The use case ends.

3.1.3 Administration User Cases



3.1.3.1 Use Case 11: View User Information

Name: View User Information
Identifier: UC11
Actors: Administrator
Description: The administrator uses the system to view information about the users.
Precondition(s):

- The administrator wishes to view user information for some purpose.

Postcondition(s):

- The user information has been displayed to the administrator.

Basic Course of Action:

1. The system presents the administrator with a list of users.
2. The administrator examines the list of users.
3. The use case ends.

3.1.3.3 Use Case 12: View/Set Maximum Number of Samples for Processing

Name: View/Set Maximum Number of Samples for Processing
Identifier: UC12
Actors: Administrator
Description: The administrator uses the system to view and/or set the maximum number of samples for processing analysis.
Precondition(s):

- The administrator wishes to view or set the maximum number of samples for processing analysis for some purpose.

Postcondition(s):

- The administrator has viewed the maximum number of samples for processing analysis and changed it if desired.

Basic Course of Action:

1. The system displays the current maximum number of samples for processing analysis.
2. The administrator examines the current maximum number of samples for processing analysis.
3. The administrator changes the maximum number of samples for processing analysis. [Alternate Course A: The Administrator does not change the maximum number of samples for processing analysis]
4. The system confirms the change and displays it to the administrator.
5. The use case ends.

Alternate Course of Action A: The Administrator does not change the maximum number of samples for processing analysis

A.3. The use case ends.

3.2 Detailed Requirements

1 External interface requirements
- 1.1 User interfaces
  - 1.1.1 The user interface should consist of web pages.
  - 1.1.2 Required Inputs
    - 1.1.2.1 Use Case 1: Reference Processing Analysis with Differential Expression Testing and Novel Gene and Transcript Discovery Workflow
      - 1.1.2.1.1 The system should require the user to select whether their re
      - 1.1.2.1.2 The system should require the user to upload one fastq file p
      - 1.1.2.1.3 The system should require the user to select whether their fa
      - 1.1.2.1.4 The system should require the user to either select a referenc
      - 1.1.2.1.5 The system should require the user to either select an annota
    - 1.1.2.2 Use Case 2: Reference Processing Analysis with Differential Expression Testing Workflow
      - 1.1.2.2.1 The system should require the user to select whether their re
      - 1.1.2.2.2 The system should require the user to upload one fastq file p
      - 1.1.2.2.3 The system should require the user to select whether their fa
      - 1.1.2.2.4 The system should require the user to either select a referenc
    - 1.1.2.3 Use Case 3: Reference Processing Analysis with Novel Gene and Transcript Discovery Workflow
      - 1.1.2.3.1 The system should require the user to select whether their re
      - 1.1.2.3.2 The system should require the user to upload one fastq file p
      - 1.1.2.3.3 The system should require the user to select whether their fa
      - 1.1.2.3.4 The system should require the user to either select a referenc
      - 1.1.2.3.5 The system should require the user to either select an annota
    - 1.1.2.4 Use Case 4: De-novo Differential Expression Testing using Trinity, Tophat, and the Cufflinks Package Workflow
      - 1.1.2.4.1 The system should require the user to select whether their re
      - 1.1.2.4.2 The system should require the user to upload one fastq file p
      - 1.1.2.4.3 The system should require the user to select whether their fa
    - 1.1.2.5 Use Case 5: De-novo Differential Expression Testing using the Trinity Package Workflow
      - 1.1.2.5.1 The system should require the user to select whether their re
      - 1.1.2.5.2 The system should require the user to upload one fastq file p
      - 1.1.2.5.3 The system should require the user to select whether their fa
    - 1.1.2.6 Use Case 6: Upload Data to the Database
      - 1.1.2.6.1 The system should require the user to upload the sequence f
      - 1.1.2.6.2 The system should require the user to select whether they w
      - 1.1.2.6.3 If the user selects Cuffdiff output data, the system should re
      - 1.1.2.6.4 If the user selects edgeR, the system should require the user
    - 1.1.2.7 Use Case 7: Query for Differentially Expressed Transcripts
      - 1.1.2.7.1 The system should required the user to select the data set to
    - 1.1.2.8 Use Case 8: Query for Differentially Expressed Genes
      - 1.1.2.8.1 The system should required the user to select the data set to

1.1.2.9 Use Case 9: Query for Novel Isoforms and Other Transcript Classifications
1.1.2.9.1 The system should required the user to select the data set to
1.1.2.10 Use Case 10: Query the BLASTable Database
1.1.2.10.1 The system should required the user to select the data set to
1.1.2.10.2 The system should required the user to either enter one or m
1.1.2.10.3 The system should require the user to specify blastn, tblastn
1.1.2.11 Use Case 11: View User Information
1.1.2.11.1 No required inputs.
1.1.2.12 Use Case 12: View/Set Maximum Number of Samples for Processing Analysis
1.1.2.12.1 No required inputs.

1.1.3 Optional Inputs

1.1.3.1 Use Case 1: Reference Processing Analysis with Differential Expression Testing and Novel Gene and Transcript Discovery Workflow
1.1.3.1.1 Non-program-specific options
1.1.3.1.1.1 The system should allow the user to specify th
1.1.3.1.2 Tophat options
1.1.3.1.2.1 The system should allow the user to upload a fi
1.1.3.1.2.2 The system should allow the user to specify the
1.1.3.1.2.3 The system should allow the user to specify the
1.1.3.1.2.4 The system should allow the user to specify the
1.1.3.1.2.5 The system should allow the user to specify the
1.1.3.1.2.6 The system should allow the user to specify the
1.1.3.1.2.7 The system should allow the user to specify the
1.1.3.1.2.8 The system should allow the user to specify the
1.1.3.1.2.9 The system should allow the user to specify the
1.1.3.1.2.10 The system should allow the user to specify the
1.1.3.1.2.11 The system should allow the user to specify the
1.1.3.1.2.12 The system should allow the user to specify the
1.1.3.1.2.13 The system should allow the user to specify the
1.1.3.1.2.14 The system should allow the user to specify the
1.1.3.1.2.15 The system should allow the user to specify wh
1.1.3.1.2.16 The system should allow the user to disable the
1.1.3.1.2.17 The system should allow the user to enable mic
1.1.3.1.3 Cufflinks options
1.1.3.1.3.1 The system should allow the user to enable mu
1.1.3.1.3.2 The system should allow the user to upload a n
1.1.3.1.4 Cuffcompare options
1.1.3.1.4.1 The system should allow the user to specify tha
1.1.3.1.4.2 The system should allow the user to specify tha
1.1.3.1.5 Cuffdiff options
1.1.3.1.5.1 The system should allow the user to specify a l
1.1.3.1.5.2 The system should allow the user to specify tha
1.1.3.1.5.3 The system should allow the user to specify tha

1.1.3.1.5.4 The system should allow the user to enable mu
1.1.3.1.5.5 The system should allow the user to set the mi
1.1.3.1.5.6 The system should allow the user to upload a n
1.1.3.1.5.7 The system should allow the user to set the FD

1.1.3.2 Use Case 2: Reference Processing Analysis with Differential Expression Testing Workflow

1.1.3.2.1 Non-program-specific options

1.1.3.2.1.1 The system should allow the user to specify th

1.1.3.2.2 Tophat options

1.1.3.2.2.1 The system should allow the user to upload a fi
1.1.3.2.2.2 The system should allow the user to specify the
1.1.3.2.2.3 The system should allow the user to specify the
1.1.3.2.2.4 The system should allow the user to specify the
1.1.3.2.2.5 The system should allow the user to specify the
1.1.3.2.2.6 The system should allow the user to specify the
1.1.3.2.2.7 The system should allow the user to specify the
1.1.3.2.2.8 The system should allow the user to specify the
1.1.3.2.2.9 The system should allow the user to specify the
1.1.3.2.2.10 The system should allow the user to specify the
1.1.3.2.2.11 The system should allow the user to specify the
1.1.3.2.2.12 The system should allow the user to specify the
1.1.3.2.2.13 The system should allow the user to specify the
1.1.3.2.2.14 The system should allow the user to specify the
1.1.3.2.2.15 The system should allow the user to specify wh
1.1.3.2.2.16 The system should allow the user to disable the
1.1.3.2.2.17 The system should allow the user to enable mic

1.1.3.2.3 Cufflinks options

1.1.3.2.3.1 The system should allow the user to enable mu
1.1.3.2.3.2 The system should allow the user to upload a n

1.1.3.2.5 Cuffdiff options

1.1.3.2.5.1 The system should allow the user to specify a l
1.1.3.2.5.2 The system should allow the user to specify tha
1.1.3.2.5.3 The system should allow the user to specify tha
1.1.3.2.5.4 The system should allow the user to enable mu
1.1.3.2.5.5 The system should allow the user to set the mi
1.1.3.2.5.6 The system should allow the user to upload a n
1.1.3.2.5.7 The system should allow the user to set the FD

1.1.3.3 Use Case 3: Reference Processing Analysis with Novel Gene and Transcript Discovery Workflow

1.1.3.3.1 Non-program-specific options

1.1.3.3.1.1 The system should allow the user to specify th

1.1.3.3.2 Tophat options

1.1.3.3.2.1 The system should allow the user to upload a fi
1.1.3.3.2.2 The system should allow the user to specify the
1.1.3.3.2.3 The system should allow the user to specify the
1.1.3.3.2.4 The system should allow the user to specify the

1.1.3.3.2.5 The system should allow the user to specify the
1.1.3.3.2.6 The system should allow the user to specify the
1.1.3.3.2.7 The system should allow the user to specify the
1.1.3.3.2.8 The system should allow the user to specify the
1.1.3.3.2.9 The system should allow the user to specify the
1.1.3.3.2.10 The system should allow the user to specify the
1.1.3.3.2.11 The system should allow the user to specify the
1.1.3.3.2.12 The system should allow the user to specify the
1.1.3.3.2.13 The system should allow the user to specify the
1.1.3.3.2.14 The system should allow the user to specify the
1.1.3.3.2.15 The system should allow the user to specify wh
1.1.3.3.2.16 The system should allow the user to disable the
1.1.3.3.2.17 The system should allow the user to enable mic

1.1.3.3.3 Cufflinks options

1.1.3.3.3.1 The system should allow the user to enable mu
1.1.3.3.3.2 The system should allow the user to upload a n

1.1.3.3.4 Cuffcompare options

1.1.3.3.4.1 The system should allow the user to specify tha
1.1.3.3.4.2 The system should allow the user to specify tha

1.1.3.4 Use Case 4: De-novo Differential Expression Testing using Trinity, Tophat, and the Cufflinks Package Workflow

1.1.3.4.1 Non-program specific options

1.1.3.4.1.1 The system should allow the user to specify t

1.1.3.4.2 Trinity options

1.1.3.4.2.1 The system should allow the user to specify t
1.1.3.4.2.2 The system should allow the user to enable ja

1.1.3.4.2 Tophat options

1.1.3.4.2.1 The system should allow the user to specify the
1.1.3.4.2.2 The system should allow the user to specify the
1.1.3.4.2.3 The system should allow the user to specify the
1.1.3.4.2.4 The system should allow the user to specify the
1.1.3.4.2.5 The system should allow the user to specify the
1.1.3.4.2.6 The system should allow the user to specify the
1.1.3.4.2.7 The system should allow the user to specify the
1.1.3.4.2.8 The system should allow the user to specify the
1.1.3.4.2.9 The system should allow the user to specify the
1.1.3.4.2.10 The system should allow the user to specify the
1.1.3.4.2.11 The system should allow the user to specify the
1.1.3.4.2.12 The system should allow the user to specify the
1.1.3.4.2.13 The system should allow the user to specify the
1.1.3.4.2.14 The system should allow the user to specify wh
1.1.3.4.2.15 The system should allow the user to disable the
1.1.3.4.2.16 The system should allow the user to enable mic

1.1.3.4.3 Cufflinks options

1.1.3.4.3.1 The system should allow the user to enable mu

1.1.3.10.1.3 The system should allow the user to specify t
1.1.3.10.1.4 The system should allow the user to specify t
1.1.3.10.1.5 The system should allow the user to specify t
1.1.3.10.1.6 The system should allow the user to specify t
1.1.3.10.1.7 The system should allow the user to specify t
1.1.3.10.1.8 The system should allow the user to specify t
1.1.3.10.1.9 The system should allow the user to specify t
1.1.3.10.1.10 The system should allow the user to specify t
1.1.3.10.1.11 The system should allow the user to specify t
1.1.3.10.2 If tblastn is selected as the BLAST query program to use, th
1.1.3.10.2.1 The system should allow the user to specify t
1.1.3.10.2.2 The system should allow the user to specify t
1.1.3.10.2.3 The system should allow the user to specify t
1.1.3.10.2.4 The system should allow the user to specify t
1.1.3.10.2.5 The system should allow the user to specify t
1.1.3.10.2.6 The system should allow the user to specify t
1.1.3.10.2.7 The system should allow the user to specify t
1.1.3.10.2.8 The system should allow the user to specify t
1.1.3.10.2.9 The system should allow the user to specify t
1.1.3.10.2.10 The system should allow the user to specify t
1.1.3.10.3 If tblastx is selected as the BLAST query program to use, th
1.1.3.10.3.1 The system should allow the user to specify t
1.1.3.10.3.2 The system should allow the user to specify t
1.1.3.10.3.3 The system should allow the user to specify t
1.1.3.10.3.4 The system should allow the user to specify t
1.1.3.10.3.5 The system should allow the user to specify t
1.1.3.10.3.6 The system should allow the user to specify t
1.1.3.10.3.7 The system should allow the user to specify t
1.1.3.10.3.8 The system should allow the user to specify t
1.1.3.10.3.9 The system should allow the user to specify t
1.1.3.11 Use Case 11: View User Information
1.1.3.11 No optional inputs.
1.1.3.12 Use Case 12: View/Set Maximum Number of Samples for Processing Analysis
1.1.3.12.1 The system should allow the administrator to specify the ma
1.1.4 Outputs
1.1.4.1 Use Case 1: Reference Processing Analysis with Differential Expression Testing and Novel Gene and Transcript Discovery Workflow
1.1.4.1.1 The system should make the Transcript differential expressi
1.1.4.1.2 The system should make the Gene differential expression (g
1.1.4.1.3 The system should make the Transcript FPKM tracking (iso
1.1.4.1.4 The system should make the fasta sequence file of the assem
1.1.4.2 Use Case 2: Reference Processing Analysis with Differential Expression Testing Workflow
1.1.4.2.1 The system should make the Transcript differential expressi

1.1.4.9.1.5 The fifth column should contain the transcrip
1.1.4.9.1.6 The sixth column should contain the coverag
1.1.4.9.1.7 The seventh column should contain the FPK
1.1.4.9.1.8 The eight column should contain the lower b
1.1.4.9.1.9 The ninth column should contain the upper b
1.1.4.9.1.10 The tenth column should contain the status fo
1.1.4.9.1.11 For any additional samples, four new column
1.1.4.10 Use Case 10: Query the BLASTable Database
1.1.4.10.1 The system should display an HTML page containing the B
1.1.4.11 Use Case 11: View User Information
1.1.4.11.1 The system should display an HTML table containing a list
1.1.4.11.1.1 The first column should contain the eID of th
1.1.4.11.1.2 The second column should contain the email
1.1.4.13 Use Case 13: View/Set Maximum Number of Samples for Processing Analysis
1.1.4.13.1 The system should display the most recent version of the ma
1.1.5 All input fields should have some kind of help message explaining them.

1.2 Hardware interfaces

1.2.1 The system's hardware should have a port to allow for Internet access.

1.3 Software interfaces

1.3.1 The system should interface with at least version 2.0.3 of the Tophat software.

1.3.2 The system should interface with at least version 1.3.0 of the Cufflinks package.

1.3.3 The system should interface with at least version 2.2.25 of the NCBI BLAST+ software.

1.3.5 The system should interface with at least version 2012-06-08 of the Trinity package.

1.3.6 The system should interface with or link to a quality filtering software that is capable of removing low quality reads and contamination from fastq or fasta files.

1.4 Communication interfaces

1.4.1 To protect the security of research data, the system should use https instead of http for all its web pages.

2 Functional requirements

2.1 The system should be able to perform all the basic courses of action and all the alternate courses of action for all the use cases specified in section 3.1: Use Cases.

3 Performance requirements

3.1 The system should be able to store up to 50 TB of data in its database.

3.2 The system should be able to handle up to 5 concurrent users.

4 Logical database requirements

4.1 The database should have the tables necessary to allow the system to satisfy the requirements.

5 Design constraints

5.1 The system should be implemented in a Linux compatible language and ecosystem.

5.2 The system should be able to run on Beocat.
5.3 The system should have a web interface.
5.4 The system should have a license that makes it free for academic use.

6 Software system attributes
- 6.1 Reliability
  - 6.1.1 The system's database should use transactions so that a transaction happens completely or not at all.
  - 6.1.2 The system's database should be backed up or replicated to at least one different hard drive than the one it resides on.
- 6.2 Availability
  - 6.2.1 The system should have at least 95% uptime.
- 6.3 Security
  - 6.3.1 The system should not allow a user to access any part of the application besides the sign-in screen until that user has successfully logged in.
  - 6.3.2 The system should encrypt all of its web pages with https.
  - 6.3.3 The system should only allow users to view and query only the data sets that they themselves have uploaded or data sets that other users have explicitly shared with them.
- 6.4 Maintainability
  - 6.4.1 The system should have tests which produce at least 80% code coverage.
- 6.5 Portability
  - 6.5.1 The system should run on the computer in the Bioinformatics center well enough to satisfy all of the requirements.
  - 6.5.2 The system should run on Beocat well enough to satisfy all of the requirements.

3.7 Other requirements
- 3.7.1 The system must have the approval of two Biologists who are proficient in RNA-seq analysis to confirm that it easy enough to use and has enough features to be useful to Biologists interested in RNA-seq Analysis.

3.3 Criticality of Requirements

To be determined.

4. References

1. BLAST, 2012. Retrieved November 3rd, 2012, from Wikipedia: http://en.wikipedia.org/wiki/BLAST.
2. Beocat. Retreived November 3rd, 2012, from Department of Computing and Information Sciences, Kansas State University: http://beocat.cis.ksu.edu/.
3. Cufflinks: Transcript assembly, differential expression, and differential regulation for RNA-Seq, 2012. Retrieved November 3rd, 2012, from the Miller Institute for Basic Research in Science at UC Berkeley: http://cufflinks.cbcb.umd.edu/index.html.
4. Fasta Format. Retrieved November 3rd from EverythingBio: http://www.everythingbio.com/glos/definition.php?word=FASTA+format.

5. The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants, 2010. Retrieved November 3rd, 2012, from National Center for Biotechnology Information: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2847217/.
6. False Discovery Rate, 2012. Retrieved on November 3rd from Wikipedia: http://en.wikipedia.org/wiki/False_discovery_rate.
7. How does Cufflinks calculate transcript abundances? Retrieved November 3rd, 2012, from the Miller Institute for Basic Research in Science, UC Berkeley: http://cufflinks.cbcb.umd.edu/howitworks.html#hqua.
8. Gene, 2012. Retrieved November 3rd, 2012, from Wikipedia: http://en.wikipedia.org/wiki/Gene.
9. An Introduction to Gene Ontology. Retrieved November 3rd, 2012, from the Gene Ontology Consortium: http://www.geneontology.org/GO.doc.shtml.
10. Gene Transfer Format, 2010. Retrieved November 3rd, 2012,from Wikipedia: http://en.wikipedia.org/wiki/Gene_transfer_format.
11. Analysis tools - BLAST - blastall examples. Retrieved November 3rd, 2012, from the Computational Biology Research Group, University of Oxford: http://www.compbio.ox.ac.uk/analysis_tools/BLAST/BLAST_blastall/blastall_examples.shtml .
12. P value. Retrieved November 3rd, 2012, from The Free Dictionary: http://medical-dictionary.thefreedictionary.com/P+value.
13. RNA-Seq, 2012. Retrieved November 3rd, 2012, from Wikipedia: http://en.wikipedia.org/wiki/RNA-Seq.
14. RNA-Seq Analysis, 2012. Retrieved November 3rd, 2012, from Wikipedia: http://en.wikipedia.org/wiki/RNA-Seq#Analysis.
15. RNA-Seq De novo Assembly Using Trinity, 2012. Retrieved November 3rd, 2012, from the Broad Institute and the Hebrew University of Jerusalem: http://trinityrnaseq.sourceforge.net/.
16. Tophat Manual. Retrieved November 3rd, 2012, from the McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University:: http://tophat.cbcb.umd.edu/manual.html.
17. Cufflinks Manual.Retrieved November 3rd, 2012, from the Miller Institute for Basic Research in Science, UC Berkeley: http://cufflinks.cbcb.umd.edu/manual.html
18. Getting Started With Cufflinks. Retrieved November 3rd, 2012, from the Miller Institute for Basic Research in Science, UC Berkeley: http://cufflinks.cbcb.umd.edu/tutorial.html.
19. Examples included with the downloaded Trinity package(trinityrnaseq_r2012-10-05) in the sample data folder, 2012. Retrieved October 29, 2012, from the Broad Institute and the Hebrew University of Jerusalem: http://sourceforge.net/projects/trinityrnaseq/files/trinityrnaseq_r2012-10-05.tgz/download .
20. Identifying Differentially Expressed Trinity Transcripts, 2012. Retrieved November 3rd, 2012, from the Broad Institute and the Hebrew University of Jerusalem: http://trinityrnaseq.sourceforge.net/analysis/diff_expression_analysis.html.
21. Command line help displayed when adding the --help option for blastn, tblastn, and tblastx, 2012. Retrieved October 29, 2012, from Ubuntu's 12.10 release (universe) apt-get repository.

5. Document History

- 11/5/2012 -- Submitted to Dr. Caragea for review