WEB-BASED TOOL FOR MULTIPLE SEQUENCE ALIGNMENT ANALYSIS

A thesis submitted to

the Faculty of Engineering and Natural Sciences of

International University of Sarajevo in partial

fulfillment of the requirements for the

degree of Bachelor of Science

in SOFTWARE ENGINEERING

by

Fejsal Perva

2022

Thesis written by

FEJSAL PERVA

Thesis Committee

Associate Professor Dr. Kanita Karađuzović-Hadžiabdić	International University of Sarajevo, Bosnia and Herzegovina, Supervisor
Associate Professor Dr. Khaldoun Al-Khalidi	International University of Sarajevo, Bosnia and Herzegovina

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

FEJSAL PERVA

INTERNATIONAL UNIVERSITY OF SARAJEVO

DECLARATION OF COPYRIGHT AND AFFIRMATION

OF FAIR USE OF UNPUBLISHED WORK

WEB-BASED TOOL FOR MULTIPLE SEQUENCE ALIGNMENT ANALYSIS

Affirmed by Fejsal Perva

Signature Date

ABSTRACT

Multiple sequence alignment (MSA) is a technique used for comparing two or more biological sequences. However, tools that perform this technique return text files which can be quite tedious to compare manually since a sequence can have thousands of nucleobases that are represented as characters in a text file. Within this work, using the files returned from Clustal Omega which performs MSA, a web application using R and its library Shiny was designed and implemented to enable easier data analysis between two or more sequences. Even though there is an application that performs similar functionalities, it requires installation and configuration which can be time-consuming. The web application computes several metrics and constructs graphs based on those metrics. In addition to that, reference lists are created and corresponding graphs are created. Those graphs can then be used for the interpretation of returned text files.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I want to thank my family and friends for supporting me through these four years of studies. Also, I want to thank my mentor, Dr. Kanita Karađuzović-Hadžiabdić, and my co-mentor, Dr. Muhamed Adilović, for helping me finish my thesis.

Fejsal Perva

14.06.2022, Sarajevo, Bosnia and Herzegov

CHAPTER ONE

INTRODUCTION

From its very beginnings, genetics has been an interesting scientific field for many researchers. It is focused on studying genes and heredity i.e., how specific traits are being passed on [1]. Genes are the underlying both physical and functioning unit of heredity and they are made up of DNA [2]. In genes during certain processes or as a result of external factors, such as the process of copying DNA, mutations which are changes in the DNA sequence might occur [3]. As a result, over time, different sequences originating from the same ancestor could appear. Eventually, different organisms originating from a single ancestor might turn up. For instance, several variants of the now widely popular coronavirus were identified [4]. However, before trying to identify similar regions between those genes, researchers use sequence alignment to arrange DNA, RNA, or protein sequences [5].

Sequence alignment of more than two DNA, RNA, or protein sequences is called multiple sequence alignment (MSA) [6]. One of the goals of MSA is to compare the subregions of these sequences with one another to detect any similarities between them [6]. The structures of biological sequences can have thousands of base pairs [7] [8]. A base pair, in turn, consists of two nucleobases which can be a combination of nucleobases adenine (A) and thymine (T) or a combination of nucleobases cytosine (C) and guanine (G) [9]. This means that biological sequences are represented as sequences of characters A, C, T, and G. Since aligning those characters would require a significant amount of computations, specific software that performs MSA is required. One of the most widely used software for MSA are Clustal series of programs [10] [11] with Clustal Omega being the latest version.

Multiple sequence alignment is one of the most frequently used techniques in bioinformatics since it provides an understanding of the relationship between sequence, structure, and functionalities of biological sequences [12]. Being one of the most frequently used techniques, it certainly requires significant effort to further make use of such technique by analysing results. Since Clustal returns a text file with nucleobases (expressed as letters A, C, T, or G) listed in more than two sequences (indicated by separate rows), it is quite difficult to do any analysis which might be required since one would need to go over the entire sequence one by one to draw any conclusions. By designing and implementing a web-based tool that will do this with only several clicks, we hope to eliminate this problem and allow geneticists and others interested in bioinformatics to focus on the already analysed data instead of wasting their time trying to go over the nucleobases one by one in the text file.

1.1. Problem statement

Besides Clustal Omega, other software such as IRMBASE, OXBENCH, PREFAB, and SABMARK are also used for MSA [11]. However, mentioned software (including Clustal Omega) do not provide any visual representation of the aligned sequences as well, but rather a sequence of characters in multiple lines which has to be looked into in more detail to draw any conclusion.

As Clustal Omega returns a .clustal file, a text file with sequence names and their corresponding sequences listed in multiple rows, those interested in analysing the difference between such sequences require a tool that will summarize their desired metrics. Cohen et al. [13] compared different strains of the hepatitis A virus by showing the number of mutations in different regions as well as finding the overall similarity. In addition to that, they found that insertions and deletions occurred only in one region. Mutation frequency and the number of mutations were used to do a comparison of the mutation rates of the human influenza viruses [14]. To do all this, they

had to create tables and calculated metrics from which they could draw some conclusions. However, instead of doing it manually, a user-friendly tool that will simplify the calculation of the desired metrics automatically might be preferred to avoid computational errors. Moreover, even if researchers had their desired metrics, they would need to use Spreadsheets or other software to create graphs from which they can draw some conclusions that might be required for their research such as identifying some patterns. All in all, these tasks might take some time to do which can be used in interpreting the results and writing the research itself.

1.2. Objectives

One of the main goals is to assist in analysing the aligned sequences that are received from Clustal programs. This includes displaying several graphs, and tables in a spreadsheet-like format that result from the analysis. Additionally, they would be able to analyse non-coding and coding sequences, specifically, given they have provided a way to distinguish the coding and noncoding parts of the sequence. The pairwise analysis of sequences will be implemented as well. Also, by having it as a web application and GUI, even the ones with no background in computer science or similar fields will be able to use this tool effectively.

1.3. Technologies used

The application will be developed using the programming language R and its library Shiny. Using Shiny, we can create interactive web applications using R syntax and easily deploy them on the web [15]. It allows simple and easy input of files, which are then processed in the R script, and at the end, resulting plots or data frames can be visualized. Additionally, libraries such as ggplot2 and dplyr will be used. Even though R has built-in plots, using ggplot2 we can customize our graphs in more detail to enhance the user experience [16]. Dplyr library provided easier data manipulation which is necessary to process and prepare data for data visualization [17]. Even

though these two libraries are not crucial to the application's operations, they simplify the code and save precious time which is needed for other parts of the application.

CHAPTER TWO

BACKGROUND

In this section, brief explanations regarding the structure of files, types of metrics, and types of graphs are listed.

After performing MSA, Clustal Omega generates a .clustal file with aligned sequences (it also generates other files but they will not be used for this application). .clustal file has sequence names at the beginning of each row and the corresponding sequences of up to 60 characters in the same row. Since each sequence has thousands of characters, this text file has numerous rows. To explain input files, the MERS virus and its strands, and its coding and non-coding regions will be taken for example. As it can be seen from Figure 2.1, MERS is the name of the sequence with MERS_1 being original, while those with numbers 2 to 10 are mutations. A gap is represented as a minus sign (-). Additionally, asterisks (*) show that there were no mutations at all at that position of a sequence.

```
CLUSTAL O(1.2.4) multiple sequence alignment


MERS_1      ------------------------------------------------CAGAACTTTGATTT
MERS_2      -----------------------------------------TCTTGCAGAACTTTGATTT
MERS_3      GATTTAAGTGAATAGCTTGGCTATCTCACTTCCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_4      GATTTAAGTGAATAGCTTGGCTATCTCACTTCCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_5      ----------------------ATCTCACTTCCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_6      GATTTAAGTGAATAGCTTGGCTATCTCACTTTCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_7      ----------------------ATCTCACTTCCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_8      GATTTAAGTGAATAGCTTGGCTATCTCACTTCCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_9      ------------TAGCTTGGCTATCTCACTTTCCCTCGTTCTCTTGCAGAACTTTGATTT
MERS_10     -----------------------------------CGTTCTCTTGCAGAACTTTGATTT
                                                          **************

MERS_1      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_2      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_3      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_4      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_5      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_6      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_7      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_8      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_9      TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
MERS_10     TAACGAACTTAAATAAAAGCCCTGTTGTTTAGCGTATTGTTGCACTTGTCTGGTGGGATT
            ************************************************************
```

Figure 2.1 Example of a .clustal file

Sequences are split into two general regions: coding and non-coding regions. Coding regions encode for protein while non-coding regions do not. However, non-coding regions provide some other information [18]. To determine the coding and non-coding regions of a sequence, a .csv file is used. An example of .csv file is shown in Figure 2.2, where Start and Stop columns represent the starting and stopping points of coding regions and everything beside it are non-coding regions. For instance, 1-232 is a non-coding region while 233-13387 is a coding region for this sequence. Even though other columns might not be empty as is the case with this one, in particular, they will not be of use. The .csv files are available to be downloaded for the genome used from the NCBI (National Center for Biotechnology Information) database, but they can also be generated manually.

#Name	Accession	Start	Stop	Strand	GeneID	Locus	Locus tag	Protein product	Length	Protein Name
		233	13387							
		13387	21468							
		233	13408							
		21410	25471							
		25486	25797							
		25806	26135							
		26047	26787							
		26794	27468							
		27544	27792							
		27807	28466							
		28520	29761							
		28716	29054							

Figure 2.2 Example of a .csv file

Using the .clustal file, a user can calculate several metrics such as similarity, mutation rate, number of mutations, number of transitions, number of transversions, transition/transversion ratio, number of insertions, and number of deletions. Similarity shows how similar are two sequences. The mutation rate shows the percentage of mutations between two sequences, while the number of mutations represents how many mutations there are between two sequences. Transition is a type

of mutation where an adenine mutated to guanine, or vice versa, or thymine mutated to cytosine, or vice versa. A transversion is a type of mutation where adenine is mutated to thymine or cytosine or vice versa, or guanine is mutated to thymine or cytosine, or vice versa. In general, transitions are mutations between more similar bases and are generally more likely to happen, while transversions are mutations between more dissimilar bases and are usually less likely to happen [19]. Insertion means that a sequence has a character instead of a gap, while deletion means that a sequence has a gap instead of a character compared to the reference sequence. By adding a .csv file, users can find these metrics separately for coding and non-coding regions.

Additionally, besides calculating metrics, reference lists using the .clustal and .csv can be constructed. A reference list shows a number of mutations at every point where the first sequence is a reference sequence. Using the .clustal file from Figure 2.1, at index 1 for reference sequence is a gap while for sequences 3, 4, 6, and 8 it is G which means that the number of mutations at index 1 is 4. An example of a reference list is shown in Figure 2.3.

Position	1	2	3	4	5	6	7	8	9	10
Mutations	0	0	4	5	7	2	0	2	1	0

Figure 2.3 An example of a reference list

CHAPTER THREE

RELATED WORKS

In this section, we will discuss similar tools for MSA analysis and applications that were developed using R with Shiny.

Even though several tools perform MSA and return aligned sequences as text [11], not many software that does some actual analysis on aligned sequences were found, let alone those that are web-based. On the other hand, there are thousands of web applications that were developed using R with Shiny for various purposes.

One of the software that analyzes MSA is JalView. JalView is a software that can do MSA editing, visualization, and analysis. When it comes to editing, a user can select sequences, insert or delete gaps, copy, move or delete sequences, remove gapped columns, etc. It also includes displaying several aligned sequences in separate rows so that users can go over nucleobases one by one, colouring the nucleotides with unique colors to better distinguish them, and displaying a bar chart underneath every column for a specified metric. It has both desktop and web versions, developed using Java and JavaScript, respectively [20].

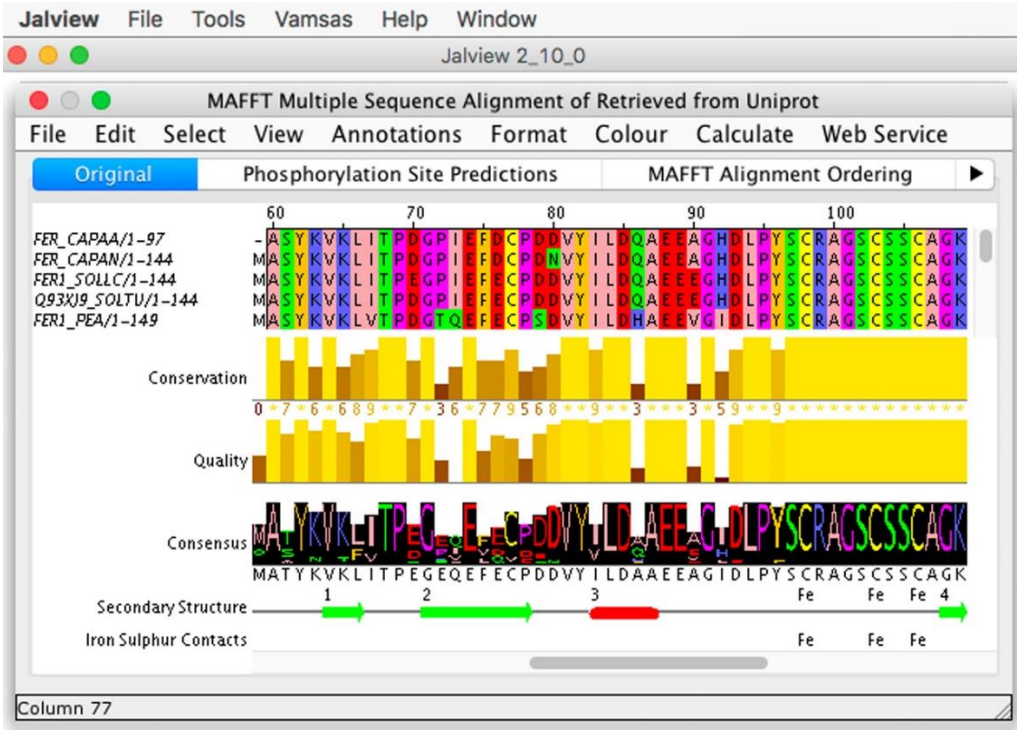

Figure 3.1 JalView UI [20]

However, even though it has a web version, a user would need to download a desktop version and install it for the web version to work. Moreover, there are no graphs that are based on an entire sequence compared to the original sequence for some specified metric e.g., similarity.

R with Shiny has been used to develop numerous web applications to suit different needs in genetics. One of the examples is a web application that can create interactive specific plots – called Circos plots that visualize genomic data [21]. The authors mention that installation and usage of similar software that can create those plots can be of issue for users who lack coding experience, which the web application does not require and therefore avoids that issue. This allows geneticists to focus on what they want to do instead of focusing on installations which can be tedious for them.

Additionally, a web application that does complete Hi-C data analysis was developed using Shiny [22]. The authors state the application integrates several packages for Hi-C data analysis

and visualization which guides the user through the entire process. Hi-C stands for high-throughput chromosome conformation capture and it is a technique that is used to study the organization of the 3D chromatin at the genome-wide level. Similar tools for this exist but they are designed for expert users. Furthermore, an application that provides the environment for population analysis has been developed [23]. As is the case with the other mentioned applications developed using Shiny, there are other software that performs population analysis. However, compared to this application they lack interactivity.

CHAPTER FOUR

SYSTEM FEATURES AND USE CASES

System features represent the software's intended functionalities and capabilities that the users will be offered to. Use cases provide explanations of how the users will accomplish their tasks and all the required steps that they need to perform that task.

4.1. System Features

All of the system's features for the web application, accompanied by the brief explanations, priority levels, user requirements, and their accompanying functional requirements are listed below.

Table 4.1 List of features

Feature number	**Feature name**
FTR 1	Input
FTR 2	Mutation analysis
FTR 3	Differential analysis of multiple sequence alignment
FTR 4	Creating a summary tables
FTR 5	Creating a reference lists
FTR 6	Tables output
FTR 7	Visualization of data

FTR 1 Input: Allowed input of .clustal and .csv files required for analysis. A Clustal file shall be used to recognize the number of sequences and store them in the appropriate data structure, while the .csv file shall be used for differential analysis of multiple sequence alignment.

Priority: Must have

UR 1.1. Input of .clustal file

FR 1.1.1. The system shall be able to accept the .clustal file as the input.

FR 1.1.2. The system shall recognize the number of sequences of the given file.

FR 1.1.3. The system shall store the file in an appropriate data structure.

UR 1.2. Optional input of .csv file

FR 1.2.1. The system shall be able to accept the .csv file as the input.

FTR 2 Mutation Analysis: The system shall analyze the differences (mutations) between sequences that have been aligned. The differences could be the difference in nucleobase or when there is a gap in one sequence compared to another.

Priority: Must have

UR 2.1. Find required metrics

FR 2.1.1. The system shall be able to find the differences in sequences pairwise, and calculate the required metrics for these sequences.

FR 2.1.2. The system shall generate a table based on the metric results.

FTR 3 Differential analysis of multiple sequence alignment Given the .csv file, the system shall be able to analyze mutations between coding and non-coding regions.

Priority: Must have

UR 3.1. Find required metrics separately for coding and non-coding regions

FR 3.1.1. The system shall divide the coding and non-coding regions of the entire sequences based on the .csv file.

FR 3.1.2. The system shall be able to analyze the differences in sequences pairwise, and calculate the required metrics for these sequences, for both coding and non-coding regions separately.

FR 3.1.3. The system shall generate a table based on the metric results.

FTR 4 Creating sequence table Display all sequences in one table.

Priority: Should have

UR 4.1. Comparison of sequences

FR 4.1.1. The system shall display all sequences in a table, one sequence per row.

UR 4.2. Download data

FR 4.1.2. The system shall allow the user to download the sequence table by clicking the appropriate button.

FTR 5 Creating a reference list Creates reference lists given the .clustal and .csv files, and metrics which can be the number of transitions, number of transversions, number of point mutations, number of gaps, or number of mutations.

Priority: Must have

UR 5.1. Creation of reference lists

FR 5.1.1. Given the .clustal and .csv files and one of the five possible metrics, the system shall create a reference list that will store the number of differences of the chosen metric at every index of the sequence.

FTR 6 Tables output Provides the way to visualize the summary table which will contain all the metrics combined. Also, the option to download the summary table will be available.

Priority: Must have

UR 6.1. Display all metrics combined in one table

FR 6.1.1. The system shall display the summary table that will contain all 9 metrics, for the entire sequence, coding, and non-coding regions resulting in 27 metrics in total.

UR 6.2. Download data

FR 6.2.1. The system shall allow the user to download the summary table by clicking a button.

FTR 7 Visualization of data Visualizes the metrics and reference lists as graphs.

Priority: Must have

UR 7.1. Visualize metrics

FR 7.1.1. The system shall display the bar chart of the entire sequence, coding, and non-coding regions when the user selects one of the nine metrics.

UR 7.2. Visualize reference lists

FR 7.2.1. The system shall display the graph that depicts the reference list of the entire sequence when the user selects one of the five metrics and the number of nucleobases per column.

FR 7.2.2. The system shall display the graph that depicts the reference list of the coding and non-coding regions when the user selects one of the five metrics and the number of nucleobases per column.

UR 7.3. Customizing elements of graphs

FR 7.3.1. The system shall allow the user to change elements of the graph such as title, x and y-axis names, and font sizes.

UR 7.4 Saving graphs

FR 7.4.1. The system shall allow the user to save graphs on their device.

4.2. Use cases

As we have already mentioned above, use cases provide explanations of how the users will accomplish their tasks and all the required steps that they need to perform that task. Use case diagram depicts a system with all its actors (types of users) and their use cases. Since the application does not have any special way of treating different users, all users will be labeled as User in the following part.

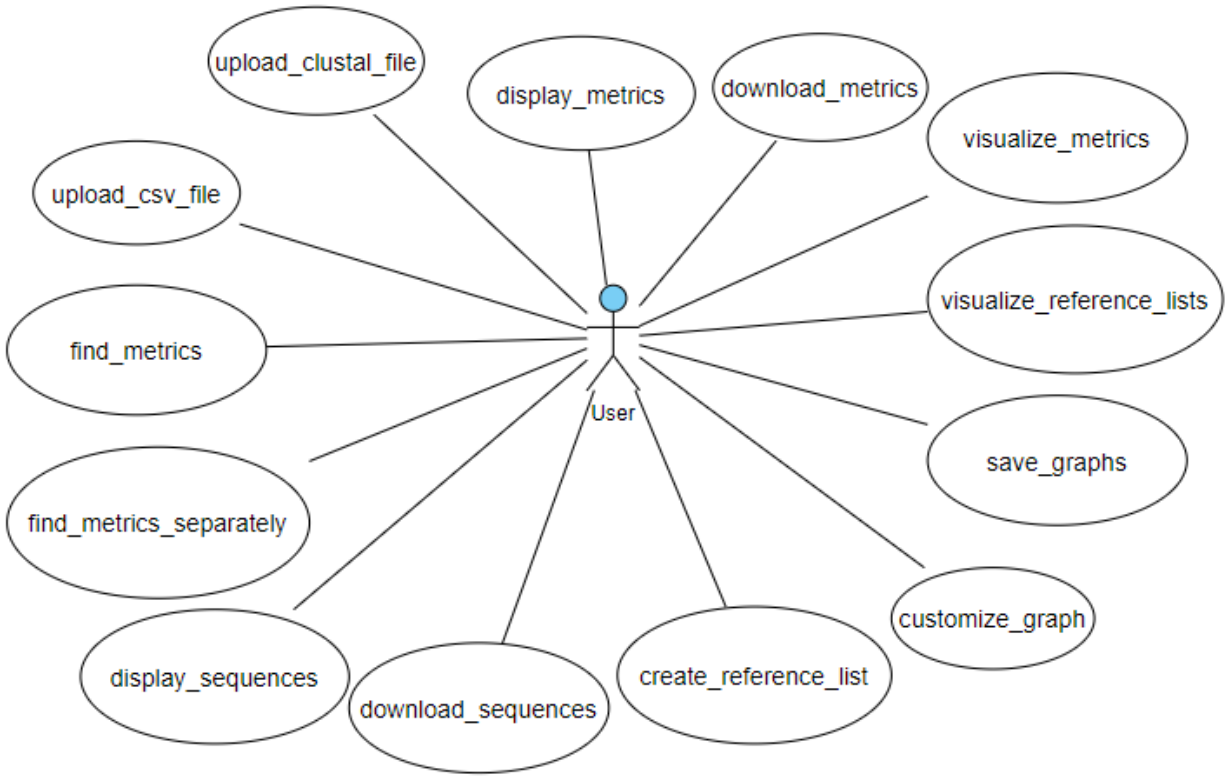

Figure 4.1 Use case diagram

As it can be seen from the use case diagram in Figure 4.1, the User can upload clustal and CSV files, find metrics for the entire sequence and separately for coding and non-coding regions, display all metrics in a summary table, display sequences, create reference lists, visualize metrics and reference lists, as well as customize those graphs. Additionally, the User can download sequences, metrics in a summary table, and graphs.

4.2.1. Detailed Use Cases

A list of the use cases as well as their explanations are listed below.

Use case ID and title: UC-1.1. Upload .clustal file

Description: A User can upload a .clustal files.

Priority: Must have

Preconditions:

1. Internet connection
2. .clustal file

Basic flow:

1. User clicks Browse... in the input field for .clustal file
2. The system opens a window for User to select their file.
3. The user selects the file
4. The system loads the file

Alternate flow: None

What can go wrong: -

Assumptions: The user has prepared data beforehand.

Use case ID and title: UC-1.2. Upload .csv file

Description: A User can upload a .csv files.

Priority: Must have

Preconditions:

1. Internet connection
2. .csv file

Basic flow:

1. The user clicks Browse... in the input field for the .csv file.
2. The system opens a window for User to select their file.
3. The user selects the file
4. The system loads the file

Alternate flow: None

What can go wrong: -

Assumptions: The user has prepared data beforehand.

Use case ID and title: UC-2.1. Find metrics for the entire sequence.

Description: A User can find the metrics for the entire sequence given the .clustal file.

Priority: Must have

Preconditions:

1. Internet connection
2. .clustal file is uploaded

Basic flow:

1. The user clicks the Process button.
2. The system processes the .clustal file and calculates metrics.

Alternate flow: None

What can go wrong: A .clustal file has issues.

Assumptions: User has uploaded .clustal file.

Use case ID and title: UC-3.1. Find metrics for coding and non-coding regions

Description: A User can find metrics for coding and non-coding regions separately.

Priority: Must have

Preconditions:

1. Internet connection
2. .clustal file is uploaded
3. .csv file is uploaded

Basic flow:

1. The user clicks on the Process button.
2. System processes the .clustal and .csv files and calculates metrics.

Alternate flow: None

What can go wrong: Issues with .clustal or .csv file.

Assumptions: None

Use case ID and title: UC-4.1. Display sequences

Description: A User can choose to display sequences in a table.

Priority: Could have

Preconditions:

1. Internet connection
2. .clustal file is uploaded

Basic flow:

1. The user chooses to display sequences.
2. The system displays sequences on the screen.

Alternate flow: None

What can go wrong: -

Assumptions: None

Use case ID and title: UC-4.2. Download sequences

Description: A User can download sequences as .csv. Each sequence will be in a separate row.

Priority: Must have

Preconditions:

1. Internet connection
2. .clustal file is uploaded.

Basic flow:

1. The user clicks the Download button for sequences.
2. The system opens a window for the user to change the file name and choose the location.
3. The user clicks the Save button.

Alternate flow: None

What can go wrong: -

Assumptions: The user has enough space on their computer.

Use case ID and title: UC-5.1. Create reference lists

Description: A User can create reference lists.

Priority: Must have

Preconditions:

1. Internet connection
2. .clustal file is uploaded
3. .csv file is uploaded

Basic flow:

1. The user clicks on the Process button.
2. System processes the .clustal and .csv files and creates reference lists

Alternate flow: None

What can go wrong: Issues with .clustal or .csv files.

Assumptions: None

Use case ID and title: UC-6.1. Display metrics

Description: A User can choose to display metrics in a table.

Priority: Could have

Preconditions:

1. Internet connection
2. Metrics are calculated.

Basic flow:

1. The user chooses to display metrics.
2. The system displays metrics on the screen.

Alternate flow: None

What can go wrong: -

Assumptions: None.

Use case ID and title: UC-6.2. Download metrics

Description: A User can download metrics that are stored in a summary table.

Priority: Must have

Preconditions:

1. Internet connection
2. Metrics are calculated for the entire sequence, and coding and non-coding regions separately.

Basic flow:

1. The user clicks the Download button for sequences.
2. The system opens a window for the user to change the file name and choose the location.
3. The user clicks the Save button.

Alternate flow: None

What can go wrong: -

Assumptions: The user has enough space on their computer.

Use case ID and title: UC-7.1. Visualize metrics

Description: A User can create graphs of metrics for the entire sequence, and coding and non-coding regions separately.

Priority: Must have

Preconditions:

1. Internet connection
2. Metrics are calculated for the entire sequence, and coding and non-coding regions separately.

Basic flow:

1. The user selects desired metric.
2. The system creates a graph of a selected metric for all sequences.
3. The system displays graphs.

Alternate flow: None

What can go wrong: -

Assumptions: None

Use case ID and title: UC-7.2. Visualize reference lists

Description: A User can create graphs of reference lists for the entire sequence, and coding and non-coding regions separately.

Priority: Must have

Preconditions:

1. Internet connection

2. Reference lists are created for the entire sequence, and coding and non-coding regions separately.

Basic flow:

1. The user selects desired metric.
2. The system creates graphs of reference lists based on the given metric.
3. The system displays graphs.

Alternate flow: None

What can go wrong: -

Assumptions: None

Use case ID and title: UC-7.3. Customize graphs

Description: A User can change parameters of graphs such as title, x and y axes titles, and text size.

Priority: Must have

Preconditions:

1. Internet connection.
2. Graphs are created.

Basic flow:

1. User changes parameters.
2. The system creates new graphs given the new parameters.
3. The system displays new graphs.

Alternate flow: None

What can go wrong: -

Assumptions: None

Use case ID and title: UC-7.4 Download graphs

Description: A User can download created graphs.

Priority: Must have

Preconditions:

1. Internet connection
2. Graphs are created.

Basic flow:

1. User right-clicks on the desired graph.
2. The user clicks on the Save graph button.
3. The system opens a window for the user to change the file name and choose the location.
4. The user clicks the Save button.

Alternate flow: None

What can go wrong: None

Assumptions: The user has enough space on their computer.

4.3. Non-functional requirements

Non-functional requirements are constraints that are put on the system. As such, they are critical to the success of the system since the implementation of non-functional requirements ensures that the software will work as intended. The non-functional requirements for the application are listed and explained in Table 4.2.

Table 4.2 List of non-functional requirements

Requirement	Description	More details
NFR1: Hardware Interface	Minimum hardware requirements for the client.	Any hardware that can run Microsoft Edge, Google Chrome, or similar.
NFR2: Communication Interface	The user needs an internet connection to access the application via a browser.	
NFR3: Availability	How often will the application be available for users?	AVL-1: The system shall be available at all times unless there are issues with hosting.

NFR4: Performance	How many minutes are needed to process a .clustal file?	PER-1: The application should process a .clustal file within 2 minutes in 90% of cases.
NFR5: Usability	How easy it is to learn how to use the application.	USY-1: Users are expected to understand how to use the application within 1 hour.
NFR6: Security	Expectations of the application regarding the security	SEC-1: Since the application is not storing anything, users will not be able to see what data has been processed already.
NFR7: Design and implementation constraints	Constraints on the design and implementation of the application.	CON-1: The application should be supported for Microsoft Edge, Google Chrome, Safari, Opera, Brave, and Mozilla Firefox.
NFR8: Internationalization and localization	Availability of the system in other languages.	IL-1 The application shall be available in English (British English).

CHAPTER FIVE

SYSTEM DESIGN

In the following two subsections, the sequence diagram and UI of the application are discussed.

5.1. Sequence diagrams

In Figure 5.1, a sequence diagram for the application is shown. It includes several use cases while the remaining use cases will be explained since they are quite similar to the ones shown in the figure. Out of 13 use cases, 11 of them are included in the sequence diagram and the remaining 2 (UC-4.2. and UC-7.4) will be briefly explained. Regarding the UC-4.2. (Download sequence), it is almost the same as UC-6.2. (Download metrics), but instead of clicking the Download button for the summary table, a User should click the Download button for the sequence table and the rest is the same. For the UC-7.4. (Download graphs), instead of clicking a Download button, the User should right-click on the graph, click Save, and after that the process is the same as for the other UC-6.2.

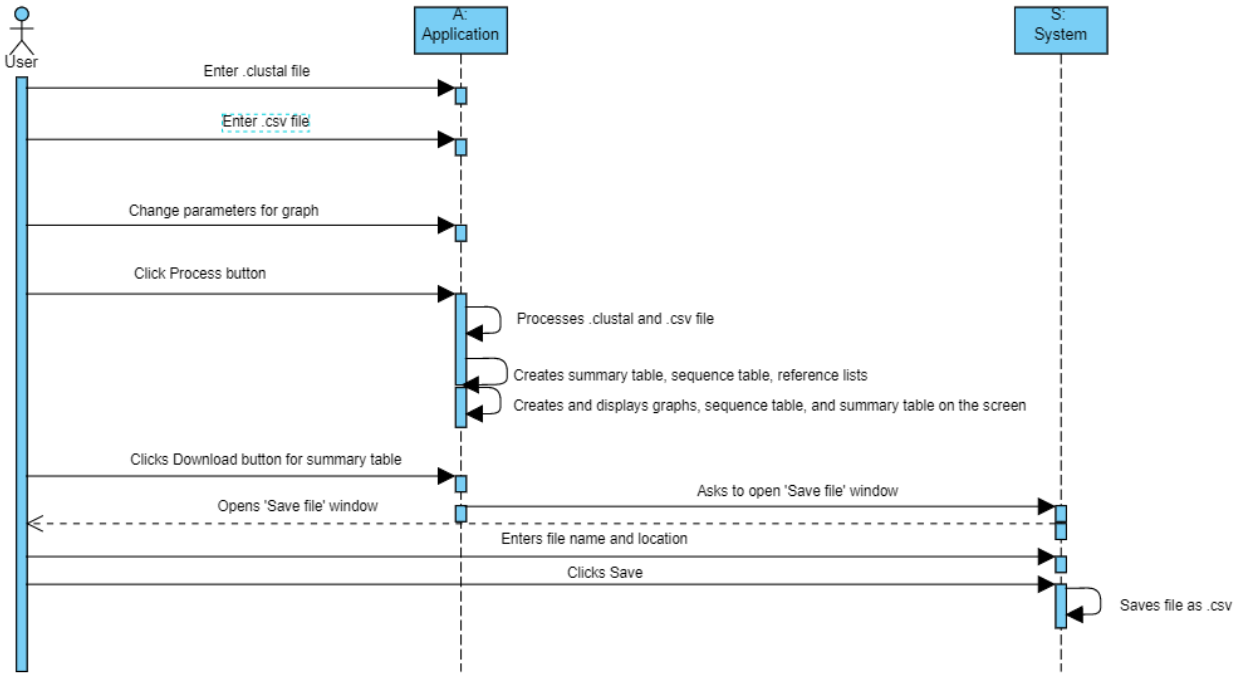

Figure 5.1 Sequence diagram for the application

5.2. User interface

The application consists of a sidebar and main sections. The sidebar contains options that affect the graphs shown in the main section. Using the figures below, the user interface accompanied by a brief explanation is explained. In Figures 5.2 and 5.3, the sidebar is shown. It contains input fields for .clustal and .csv files, as well as options to change parameters for graphs. Those parameters include title, x-axis and y-axis names, font sizes, desired metrics for graphs, as well as a number of nucleobases per bin for histograms for reference lists for the entire sequence. The higher the number of nucleobases per bin, the fewer bins there are. Additionally, the Process and Download buttons are positioned here.

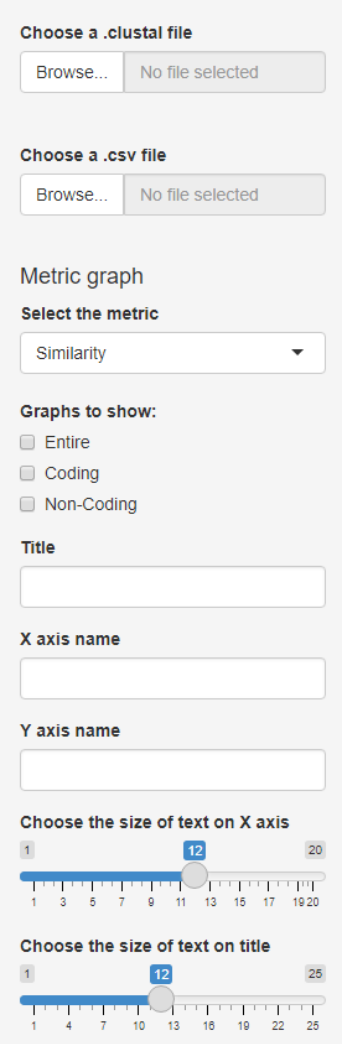

Figure 5.2 Sidebar - input of files, parameters for graph for metrics

Moreover, parameters for the other two graphs are included in the sidebar, however, since they have more or less the same options, they are not included.

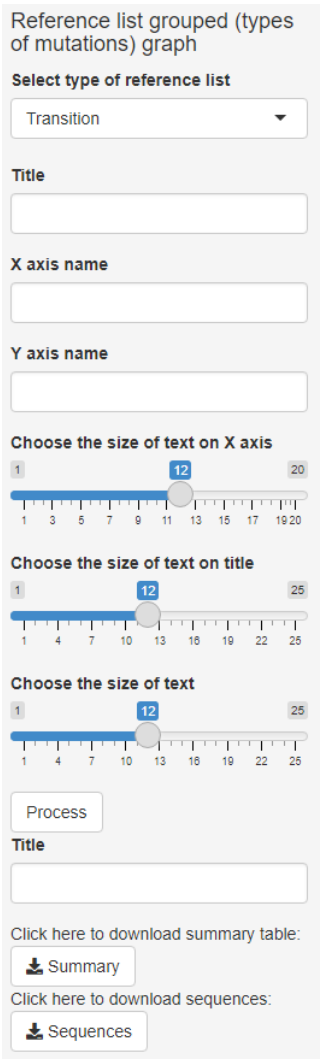

Figure 5.3 Sidebar - parameters for reference list grouped (types of mutation) graph, Process, and Download buttons

When the user uploads data, files are processed and the graphs displaying selected metric for all strands are constructed and displayed as shown in Figure 5.4. Using the parameters located in the sidebar, the user can change size and text as they prefer.

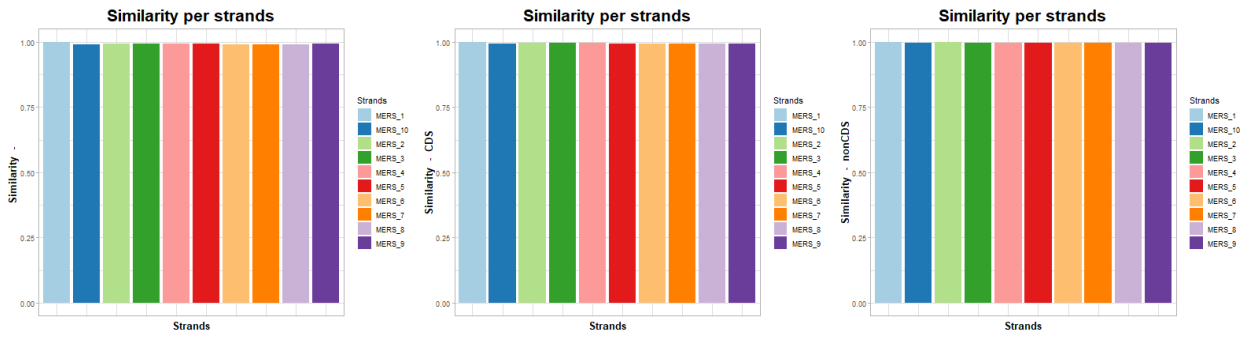

Figure 5.4 Graphs of similarity metric over different strands for the entire sequence (left), CDS regions (middle), and nonCDS regions (right)

Besides the metric graphs, a graph regarding the reference lists for one of the five metrics is constructed and displayed as well. In Figure 5.5, the reference list for the number of mutations for the entire sequence is shown. The figure shows, that most mutations occur at the beginning and at the end of the sequence.

Figure 5.5 Reference list for the entire sequence

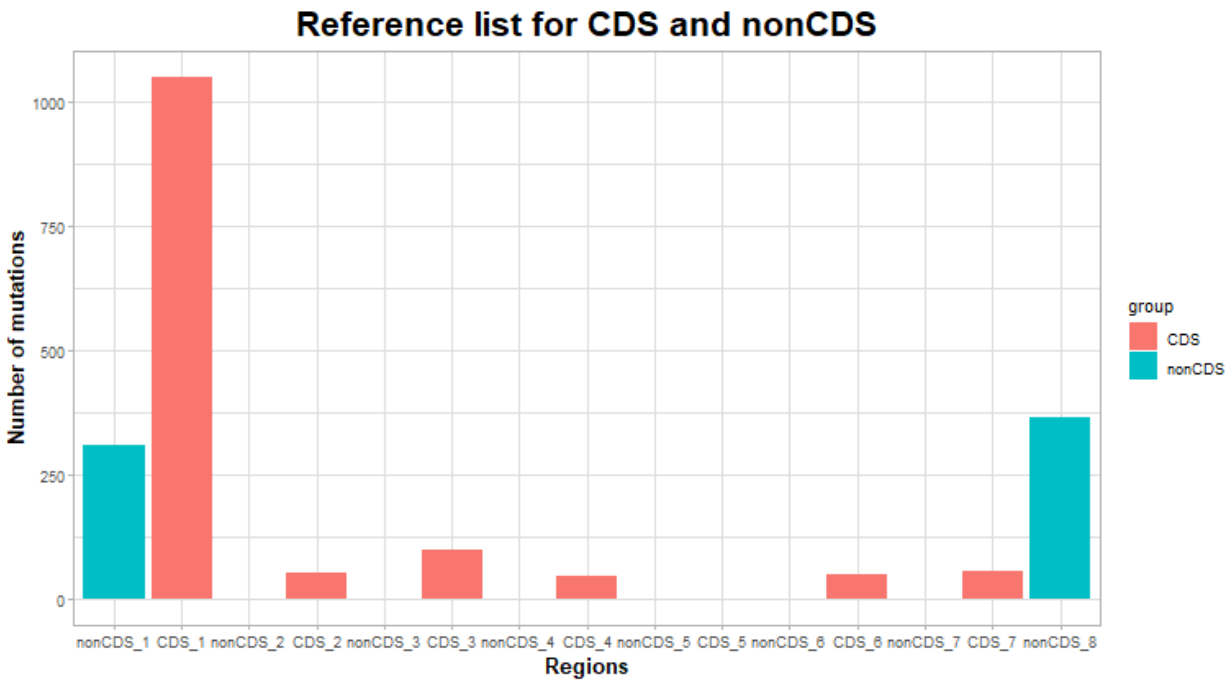

Figure 5.6 Reference list grouped

In Figure 5.6, the reference list for coding and non-coding regions is shown. The selected metric is the number of transitions. From the figure, we can deduce that the number of transitions occurs by far the most in the first coding region.

Finally, in Figure 5.7, types of mutations in different coding and non-coding regions are shown. Using the figure, it can be seen that most of the mutations are transitions, followed by gaps and transversions, respectively.

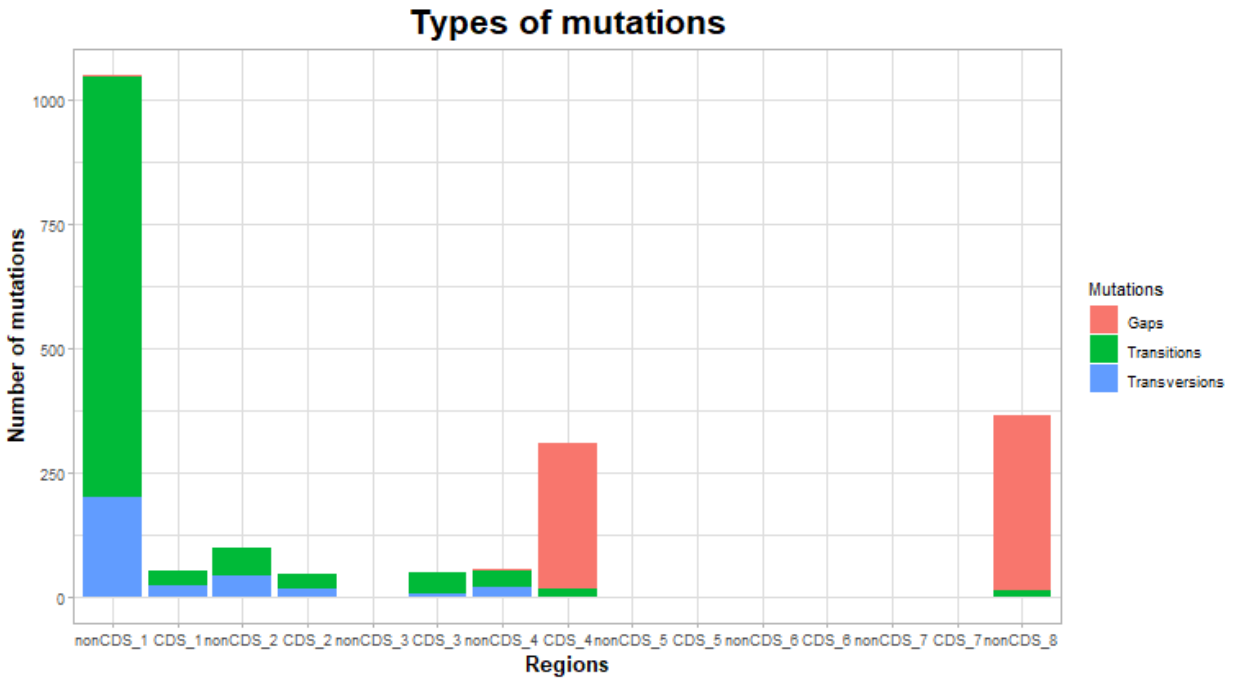

Figure 5.7 Types of mutations per region

When everything is loaded, the application looks as it is displayed in Figure 5.8.

Figure 5.8 UI of the entire application

CHAPTER SIX

CONCLUSION

The web application was designed and implemented using R and its additional libraries - Shiny was used to create the web application while other libraries helped in processing and visualization. It can assist geneticists, as well as others interested in this field, in analyzing the obtained .clustal files (which are text files with sequences being structured in rows) from the software Clustal Omega which is the most widely used software for multiple sequence analysis. Besides analyzing the entire sequence, the option to add a .csv file that would separate the sequence into coding and non-coding regions was implemented. The application then uses these files to compute metrics and using those metrics, it constructs graphs that could be of use for research. Moreover, the application creates reference lists and their corresponding graphs for the entire sequence, and coding and non-coding regions separately. Finally, for those interested in details regarding metrics, the ability to download metrics was implemented.

The application was designed in a way to not give the user much freedom, but still provides the user functionalities they need. This was done to prevent any unwanted actions.

Concerning future work, the application could be improved to support files of other software that perform MSA other than Clustal Omega. Additionally, other metrics might be computed, and/or other types of graphs might be added to improve the data analysis of those sequences.

REFERENCES

[1] "National Institute of General Medical Sciences (NIGMS)," *National Institute of General Medical Sciences (NIGMS)*. https://nigms.nih.gov/ (accessed Jun. 10, 2022).

[2] "What is a gene?: MedlinePlus Genetics." https://medlineplus.gov/genetics/understanding/basics/gene/ (accessed Jun. 10, 2022).

[3] "Mutation," *Genome.gov*. https://www.genome.gov/genetics-glossary/Mutation (accessed Jun. 10, 2022).

[4] "COVID Variants: What You Should Know," Apr. 08, 2022. https://www.hopkinsmedicine.org/health/conditions-and-diseases/coronavirus/a-new-strain-of-coronavirus-what-you-should-know (accessed Jun. 10, 2022).

[5] A. D. Prjibelski, A. I. Korobeynikov, and A. L. Lapidus, "Sequence Analysis," in *Encyclopedia of Bioinformatics and Computational Biology*, S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, Eds. Oxford: Academic Press, 2019, pp. 292–322. doi: 10.1016/B978-0-12-809633-8.20106-4.

[6] M. Y. Sofi, A. Shafi, and K. Z. Masoodi, "Chapter 6 - Multiple sequence alignment," in *Bioinformatics for Everyone*, M. Y. Sofi, A. Shafi, and K. Z. Masoodi, Eds. Academic Press, 2022, pp. 47–53. doi: 10.1016/B978-0-323-91128-3.00011-2.

[7] S. Raskin, "Genetics of COVID-19," *J. Pediatr. (Rio J.)*, vol. 97, no. 4, pp. 378–386, 2021, doi: 10.1016/j.jped.2020.09.002.

[8] F. FENNER, P. A. BACHMANN, E. P. J. GIBBS, F. A. MURPHY, M. J. STUDDERT, and D. O. WHITE, "Structure and Composition of Viruses," *Vet. Virol.*, pp. 3–19, 1987, doi: 10.1016/B978-0-12-253055-5.50005-0.

[9] "DNA and Proteins," *Genetics Generation*. https://knowgenetics.org/dna-and-proteins/ (accessed May 16, 2022).

[10] J. D. Thompson, Toby. J. Gibson, and D. G. Higgins, "Multiple Sequence Alignment Using ClustalW and ClustalX," *Curr. Protoc. Bioinforma.*, vol. 00, no. 1, p. 2.3.1-2.3.22, 2003, doi: 10.1002/0471250953.bi0203s00.

[11] R. C. Edgar and S. Batzoglou, "Multiple sequence alignment," *Curr. Opin. Struct. Biol.*, vol. 16, no. 3, pp. 368–373, Jun. 2006, doi: 10.1016/j.sbi.2006.04.004.

[12] V. Simossis, J. Kleinjung, and J. Heringa, "An Overview of Multiple Sequence Alignment," *Curr. Protoc. Bioinforma.*, vol. 3, no. 1, p. 3.7.1-3.7.26, 2003, doi: 10.1002/0471250953.bi0307s03.

[13] J. I. Cohen, J. R. Ticehurst, R. H. Purcell, A. Buckler-White, and B. M. Baroudy, "Complete nucleotide sequence of wild-type hepatitis A virus: comparison with different strains of hepatitis A virus and other picornaviruses.," *J. Virol.*, vol. 61, no. 1, pp. 50–59, Jan. 1987.

[14] "Comparison of the Mutation Rates of Human Influenza A and B Viruses." https://journals.asm.org/doi/epub/10.1128/JVI.80.7.3675-3678.2006 (accessed Jun. 01, 2022).

[15] "Shiny." https://shiny.rstudio.com/ (accessed Apr. 22, 2022).

[16] "Create Elegant Data Visualisations Using the Grammar of Graphics." https://ggplot2.tidyverse.org/ (accessed Apr. 22, 2022).

[17] "A Grammar of Data Manipulation." https://dplyr.tidyverse.org/ (accessed Apr. 22, 2022).

[18] "Non-Coding DNA," *Genome.gov*. https://www.genome.gov/genetics-glossary/Non-Coding-DNA (accessed Jun. 11, 2022).

[19] T. H. Jukes, “Transitions, transversions, and the molecular evolutionary clock,” *J. Mol. Evol.*, vol. 26, no. 1, pp. 87–98, Nov. 1987, doi: 10.1007/BF02111284.
[20] A. M. Waterhouse, J. B. Procter, D. M. A. Martin, M. Clamp, and G. J. Barton, “Jalview Version 2--a multiple sequence alignment editor and analysis workbench,” *Bioinformatics*, vol. 25, no. 9, pp. 1189–1191, May 2009, doi: 10.1093/bioinformatics/btp033.
[21] Y. Yu, Y. Ouyang, and W. Yao, “shinyCircos: an R/Shiny application for interactive creation of Circos plot,” *Bioinformatics*, vol. 34, no. 7, pp. 1229–1231, Apr. 2018, doi: 10.1093/bioinformatics/btx763.
[22] L. Di Filippo, D. Righelli, M. Gagliardi, M. R. Matarazzo, and C. Angelini, “HiCeekR: A Novel Shiny App for Hi-C Data Analysis,” *Front. Genet.*, vol. 10, 2019, Accessed: Jun. 01, 2022. [Online]. Available: https://www.frontiersin.org/article/10.3389/fgene.2019.01079
[23] N. G. Criscuolo and C. Angelini, “StructuRly: A novel shiny app to produce comprehensive, detailed and interactive plots for population genetic analysis,” *PLOS ONE*, vol. 15, no. 2, p. e0229330, Feb. 2020, doi: 10.1371/journal.pone.0229330.