**DATABASE MANAGEMENT FOR AN EFFICIENT STORAGE MODEL OF 16S METAGENOMICS DATA**

For Partial Fulfilment of M.Sc. (Bioinformatics)



By

**WAHIDUL A. BARBHUIYA**

**SEAT NO: - 2320900010**

Under the guidance of

**DARSHIT PATEL**

**Founder Centenarians Life Sciences. Pvt. Ltd**

**Department of Bioinformatics**

**Rajiv Gandhi Institute of I.T & Biotechnology**

**Bharti Vidyapeeth Deemed to Be University, Pune-46**

**April 2023**

**CERTIFICATE**



****

**To Whom It May Concern**

This is to certify that this project dissertation titled, **“Database management system for an efficient storage model of huge 16S metagenomics data**”, submitted by **Wahidul Alam Barbhuiya** (Seat No.: 2320900010) for the partial fulfillment of M.Sc. (bioinformatics) degree of Bharat Vidyapeeth Deemed University, Pune under the guidance of Mr Darshit Patel, Founder, Centenarians Life Sciences. Pvt. Ltd. This project work has been carried out during Novermber end 2022 to April 2023. Neither this review dissertation nor any part of it has been submitted for any degree or diploma elsewhere.

……………………………. ………….................................. ……………………………

( ) ( ) (Dr. G. D. Sharma)

External Examiner Internal Examiner Principal

RGITBT, BVDU, Pune-46

**Declaration & Undertaking**

I, hereby solemnly declare that the project work entitled “**Database management system for an efficient storage model of huge 16S metagenomics data**” submitted to Bharti Vidhyapeeth Deemed to Be University, Pune-46, towards the partial fulfilment of Master of Science in Bioinformatics, is the result of research work carried out by me under the guidance of Mr. Darshit Patel Centenarians Life Sciences. Pvt. Ltd.

I further declare that the work reported in this project has not been submitted previously for the award of any degree, diploma, or any other work of a similar title.

Project Guide:

Darshit Patel

Date: 26/04/2023

Place: Pune

**Acknowledgement:**

I express my sincere thanks to Rajiv Gandhi Institute of Information Technology and Biotechnology for providing the guidance for completing the project. I specially want to thank you our principal Dr. Shamim Shaikh and Assistant Professor Pritee Chunarkar-Patil for kind advice and supporting me throughout this project.

I would like to thank my guide Mr. Darshit Patel and my internal guide Sai Hariharan, under whom working was a privilege and a great experience. I also thank my colleagues especially my senior Jagan. L for helping me during this work. I am grateful because of his continuous motivation and guidance. I am also grateful to all my classmates for their support.

Last but not the least, I am thankful to my parents for their encouragement and moral support throughout the course, all of which made the completion of project work possible. In addition, my great gratitude goes to my parents. Thank you for your unconditional love and unshakeable support all the way throughout my studies.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Topic** | **Page No.** |
| 1  1.1  1.2  1.3  1.4  1.5 | **Introduction**  Introduction  Metagenomics and sequencing technique  16S metagenomics  16S taxonomic abundance calculation  Alpha diversity index | **8**  9  10  10  11  11 |
| **2**  2.1.  2.1.1  2.1.2  2.1.3  2.1.3.1  2.2  2.3.1  2.4.1  2.4.2  2.4.3  2.4.3.1  2.4.4  2.5  2.6  2.7 | **Methods**  Methods  Front end  Backend  Database  Mysql  Basic flow of the software  DBMS and flask workflow diagram  Data submission portal  Pseudo code  Condition for uploading file  Directory folder for zip format  Methods for uploading files  Output  Flask integration  Main.py | **12-42**  13  13  13  13  14-15  16  17  18  18-36  36  36  36  36-40  41-42  42 |
| **3**  3.1  3.2  3.3 | **Results and Discussion**  Results  Added feature  Discussion | **43-50**  44-45  45-49  50 |
| **4**  4.1 | **Conclusion**  Conclusion | **51-52**  52 |
| **5**  5.1 | **Bibliography**  Bibliography | **52-57**  53-57 |

**ABSTRACT**

This database management system is designed to provide an efficient and user-friendly interface for managing 16S metagenomics data. By incorporating Flask and MySQL, the system offers a scalable and secure solution for storing and accessing data. The pie charts and bar plots allow users to quickly analyze and interpret data, enabling them to make informed decisions based on the results.

This project can be further improved by implementing additional features such as data filtering, statistical analysis tools, and report generation capabilities. Overall, this database management system has the potential to be a valuable tool for researchers and scientists working in the field of 16S metagenomics.

**Keywords:***Database management system (DBMS),16S metagenomics data, Flask, Mysql, pie charts, bar plots*

**CHAPTER 1**

**INTRODUCTION**

**1.1 Introduction:**

Metagenomics has revolutionized the field of microbiology and is now widely used for studying microbial communities in various environments. The technique involves the sequencing of all the genetic material present in a sample, without the need for cultivation, isolation, or cloning of individual microbes. This approach generates vast amounts of data that need to be efficiently managed and analyzed to extract meaningful insights [1].

In this project, the objective is to build a database management architecture that can serve as a connecting bridge between lab management and taxonomic analysis for efficient storage of huge 16S metagenomics data. The architecture will be designed to handle large volumes of data generated from high-throughput nanopore sequencing technologies, ensuring data integrity, security, and accessibility.

The significance of this project lies in the fact that efficient storage and analysis of metagenomics data is critical for advancing our understanding of microbial diversity and its role in various environments. This architecture will not only enable researchers to manage and analyze large-scale metagenomics data but also facilitate the integration of data of different samples for comparative analysis.

Overall, this project aims to provide a comprehensive solution for managing and analyzing huge metagenomics datasets, thereby advancing the field of microbiology and contributing to our understanding of the microbial world.

This data management architecture consists of several components-

* A database to manage the metadata, lab data and 16S taxonomic measurement data.
* A portal of field submission for laboratory data and taxonomic file upload.
* Data searching and visualization [2].

**1.2 Metagenomics and sequencing technique:**

Metagenomics can be defined as the study of genetic materials which are coming directly as environmental sample.

Due to the advancement of sequencing techniques, metagenomics has gained popularity. Sequencing is the process of determining the nucleotide order in a small targeted genomic region or an entire genome. Over time, sequencing methods have evolved from classical methods to next-generation sequencing methods, resulting in significant technological advancements. These advancements have greatly enhanced the value and impact of sequencing technology [4].

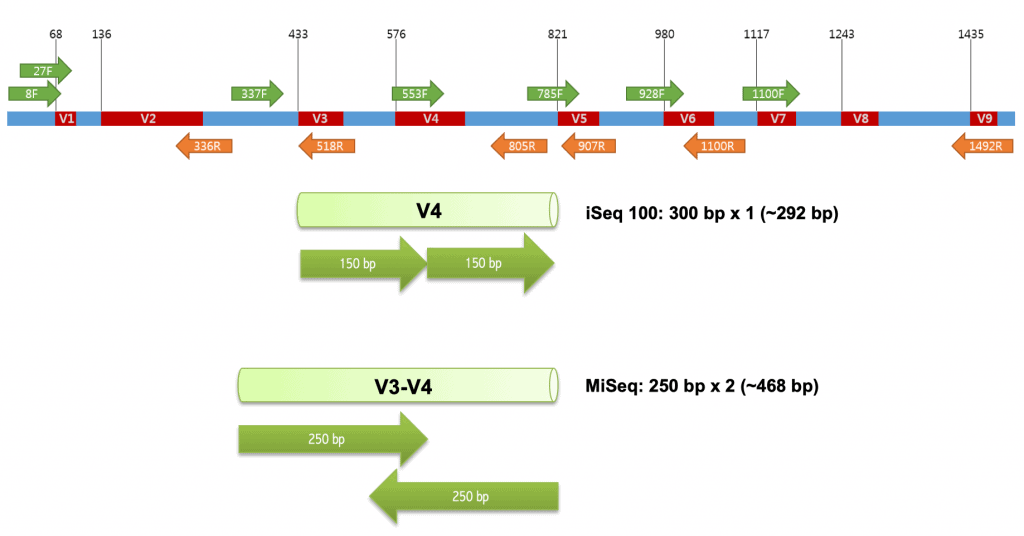
**1.3 What is 16S metagenomics?**

The 16S bacterial region refers to a highly conserved segment of the bacterial ribosomal RNA

(rRNA) gene that is approximately 1,500 base pairs in length. This region is commonly used

as a genetic marker to identify and classify bacterial species. The 16S rRNA gene contains

both variable and conserved regions that allow for phylogenetic analysis.



16S bacterial region is divided into two parts one is conserved region and other is variable region. There are nine highly conserved region and nine hyper variable regions. Conserved regions serve as “anchor” for designing PCR (polymerase chain reaction) primers and variable region is for phylogenetic study [2].

**1.4 16S Taxonomic Abundance Calculation:**

16S taxonomic abundance calculation involves analysing the relative abundance of different bacterial taxa based on their 16S rRNA gene sequences. This can be done using various bioinformatics tools and pipelines that process the raw sequence data and assign the reads to specific taxonomic groups.

**1.5 Alpha Diversity Index:**

Alpha diversity is a measure of diversity within a single sample or population, and can be calculated using various metrics. Here are some commonly used metrics for alpha diversity calculation:

* + 1. **Richness:** This is the simplest metric and represents the number of different taxa or species present in a sample.
    2. **Shannon index:** This metric takes into account both the richness and evenness of the community, and is calculated as H' = -∑(pi \* ln pi), where pi is the proportion of individuals in the ith taxon and ln is the natural logarithm.
    3. **Simpson index:** This metric measures the probability that two randomly selected individuals in the sample belong to the same taxon, and is calculated as D = ∑(ni / N)², where ni is the number of individuals in the ith taxon and N is the total number of individuals.

**CHAPTER 2**

**METHODS**

**2.1 Methods:** The methods used for developing this tool can be broadly divided into 3 categories namely: Front-end, Back-end and Database.

**2.1.1 Front-end:** Front end is the visual aspects of the website. Basic skeleton of the front-end is created by HTML (Hyper Text Markup Language), CSS (Cascading Style Sheets) and JavaScript.

The external look and aesthetics of the software can be maintained using different CSS style content [6].

JavaScript is a scripting language which is a common bridge for solving both front-end and back-end whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously and alter the document content that is displayed [6].

**2.1.2 Back-end:** Back end is focused on the server side of web application which is responsible for processing requests, handling data storage and performing other tasks that are not visible to the user.

For processing data coming from front-end that are supposed to transfer to the database or to get any outcome from there, different frame-works are used in a software development according to their requirement. Flask is one of them. The reason for me to choose flask is that it’s very lightweighted and very scalable. Since a major part of this work is related to the database, flask is the best for frame working data to the database. Flask provides a simple yet powerful interface for handling HTTP requests and responses, making it easy to build RESTful APIs and web services.

When building a Flask app, the backend code typically consists of Python functions that are mapped to specific URLs or endpoints. These functions process incoming requests and return responses to the client. Flask provides a range of tools and libraries for working with databases, handling authentication and authorization, and implementing other backend functionality.

**2.1.3 Database:** Database is an organized collection of data. It is used to support internal operations of organizations and to underpin online interactions. The database management system used in development of this tool is MySQL. Flask has flexible python build module SQLAlchemy. This library provides a powerful set of tools for creating and managing database schemas, executing queries, and performing other common database tasks. Some common features of SQLAlchemy for using any database integration things:

1. **Building web applications:** SQLAlchemy can be used to power the backend of web applications, providing an easy-to-use interface for interacting with a database. Flask and Django are two popular Python web frameworks that integrate well with SQLAlchemy.
2. **Data analysis and manipulation:** SQLAlchemy can be used to interact with large datasets stored in a database, making it a popular choice for data analysis and manipulation tasks.
3. **Object-oriented programming:** SQLAlchemy provides a powerful object-oriented interface for working with databases, allowing developers to map database tables to Python classes and work with database rows as Python objects.
4. **Testing and debugging:** SQLAlchemy provides a set of tools for testing and debugging database applications, making it easy to create and manage test databases and track down performance issues.

**2.1.3.1 MySQL:** MySQL is a popular open-source relational database management system (RDBMS) that has been around for over two decades. It is widely used by developers and businesses worldwide and has many benefits that make it a great choice for storing data, including:

**Scalability:** MySQL can handle large amounts of data and has excellent performance when it comes to read-heavy workloads. It can also scale horizontally by adding more nodes to a cluster, making it a good choice for growing applications.

**Flexibility:** MySQL supports a wide range of data types and can handle complex queries. It also has a rich set of features and tools, including triggers, stored procedures, and views, which can help simplify database administration.

**Reliability:** MySQL is known for its stability and reliability, with a proven track record of powering some of the world's most critical applications. It also has strong data protection features, such as encryption and role-based access control.

**Community support:** MySQL has a large and active community of developers and users who contribute to its development and provide support and resources to others.

**Integration:** MySQL integrates well with many programming languages, including Python, making it a good choice for web applications and other software projects.

Overall, MySQL is a reliable, flexible, and scalable database management system that can handle a wide range of applications and workloads. Its popularity and large community make it easy to find resources, support, and expertise when needed.

**2.2 Basic flow of the software:**

**START**

**Basic information of the complete tool and its applications.**

**Home Page**

**Dashboard**

**Sign in/ Sign up**

**Data submission**

**A status table for submitted data from Sample collection to taxonomy file upload will be there which are tracking by barcode. Status will appear Completed or incomplete based on the data availability in a particular section.**

**Barcode Generation**

**Barcode**

**Barcode**

**Sample collection**

**Taxonomy**

**Sample Collection**

**DNA Extraction**

**Library Preparation**

**Barcode**

Data related to this barcode

Data related to this barcode

Data related to this barcode

Data related to this barcode

**DNA Extraction**

**completed**

**incomplete**

**Barcode**

**completed**

**incomplete**

**Library Preparation**

**completed**

**incomplete**

**Barcode**

**incomplete**

**completed**

**Taxonomy**

Submit

**Dashboard**

**Submitting portal**

Javascript 🡪 🡪flask

**Batch submission**

Using ajax js

**MySQL Database**

**7 tables:**

bacteria

dna\_extraction

library\_prep

metadata

sample\_collection

score

user

**Flask end-point**

**Dashboard**

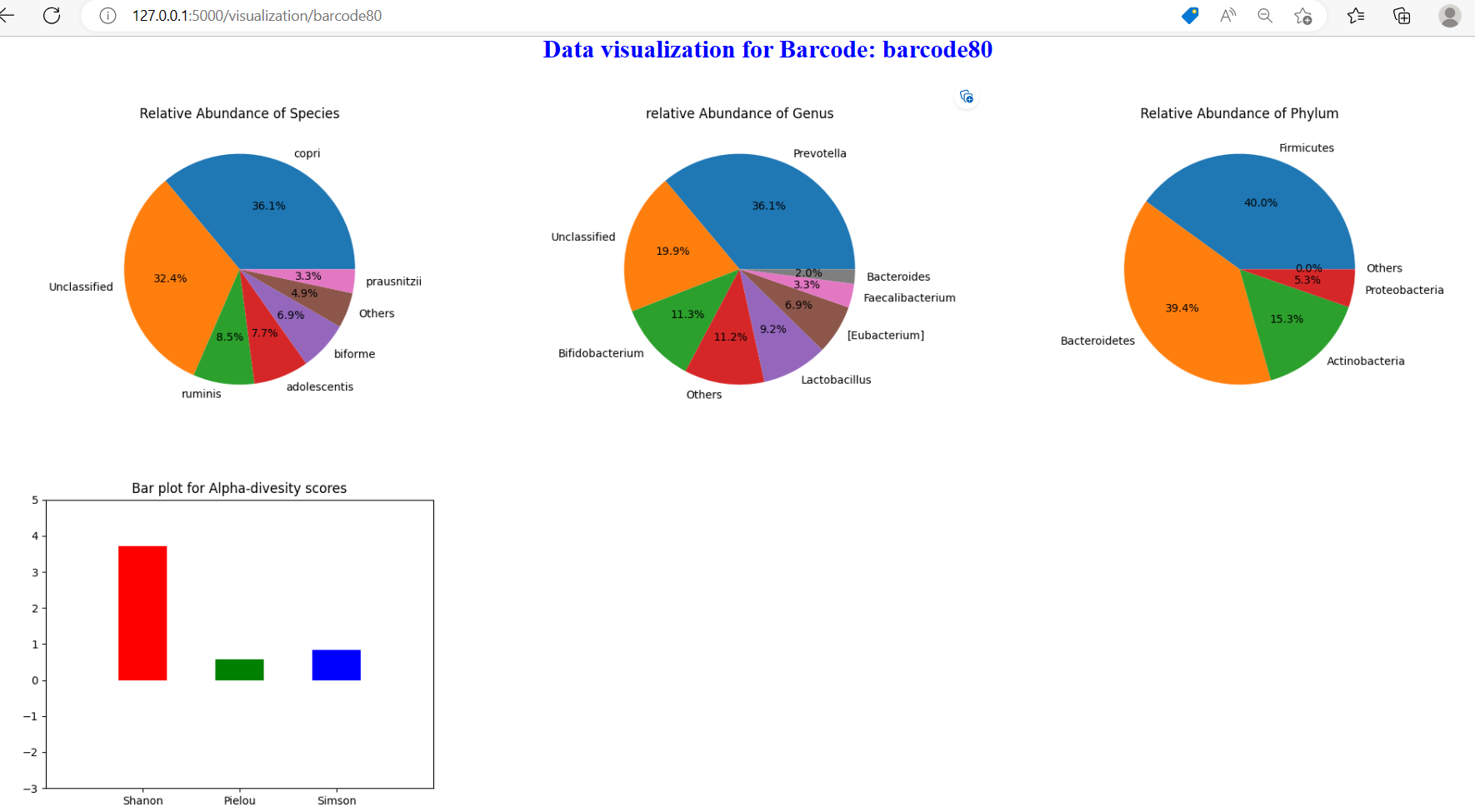
**Output Panel and Results**

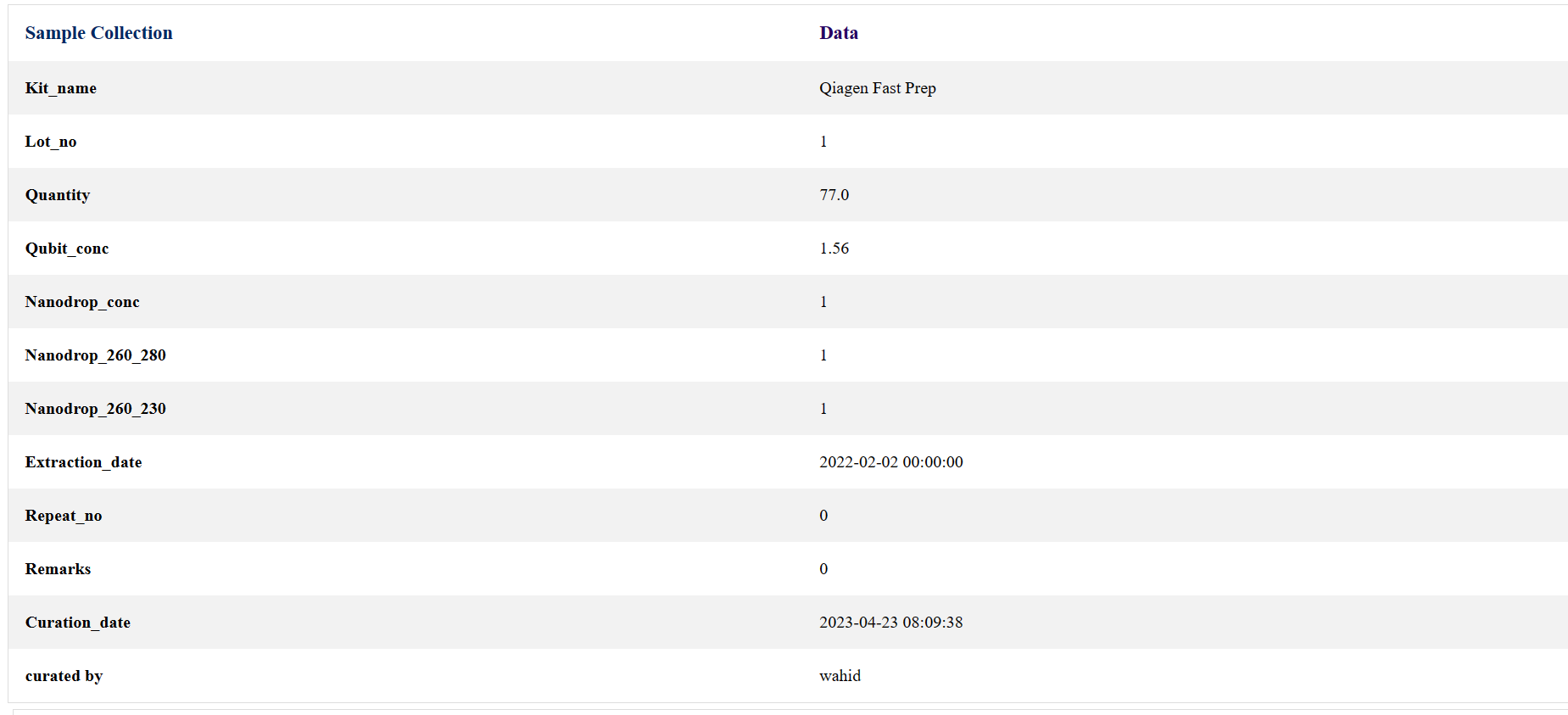
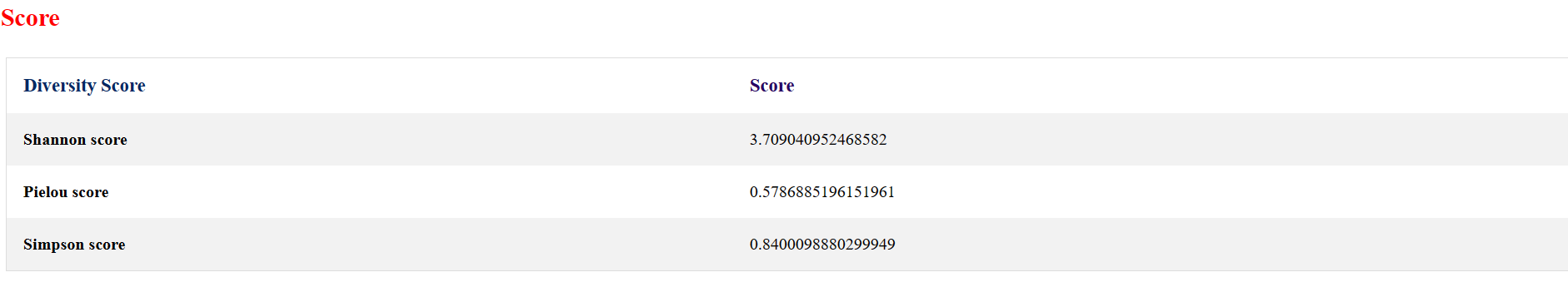
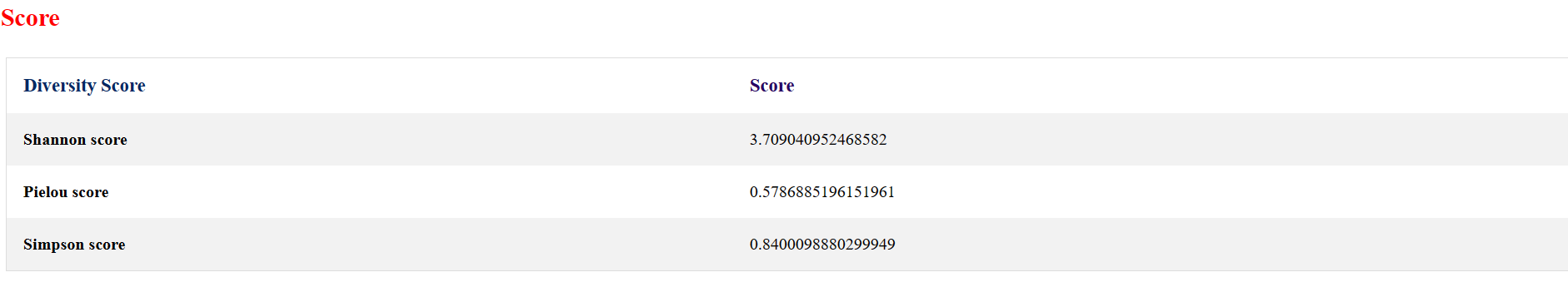
**Login page**

**Registration page**

**Visualization**

**Report**





**Users**

Sign up / Sign in

**2.4 Workflow description:**

**2.4.1 Data submission portal:**

This portal is made for data submission following a proper workflow. Workflow is made where data are taken from laboratory information related. Then there is an upload section which submits some files related to taxonomic information.

There is a detailed description how these data are related to each other and how this submission is proceeded.

Data submission process is taken in a single html page of multi-step form. Generally, a client form submission has basic things like there will be some field that are meant for filling by the client and a submitting button for sending data to the backend. For this project, there is an html form having 5 different sections. Each section is dedicated to a specific related data which are sent in a batch format. For the batch submission, ajax JavaScript function has been implemented, which sends data to each section separately, as a result of which, data will be manipulated differently.

**2.4.1.1 AJAX:**

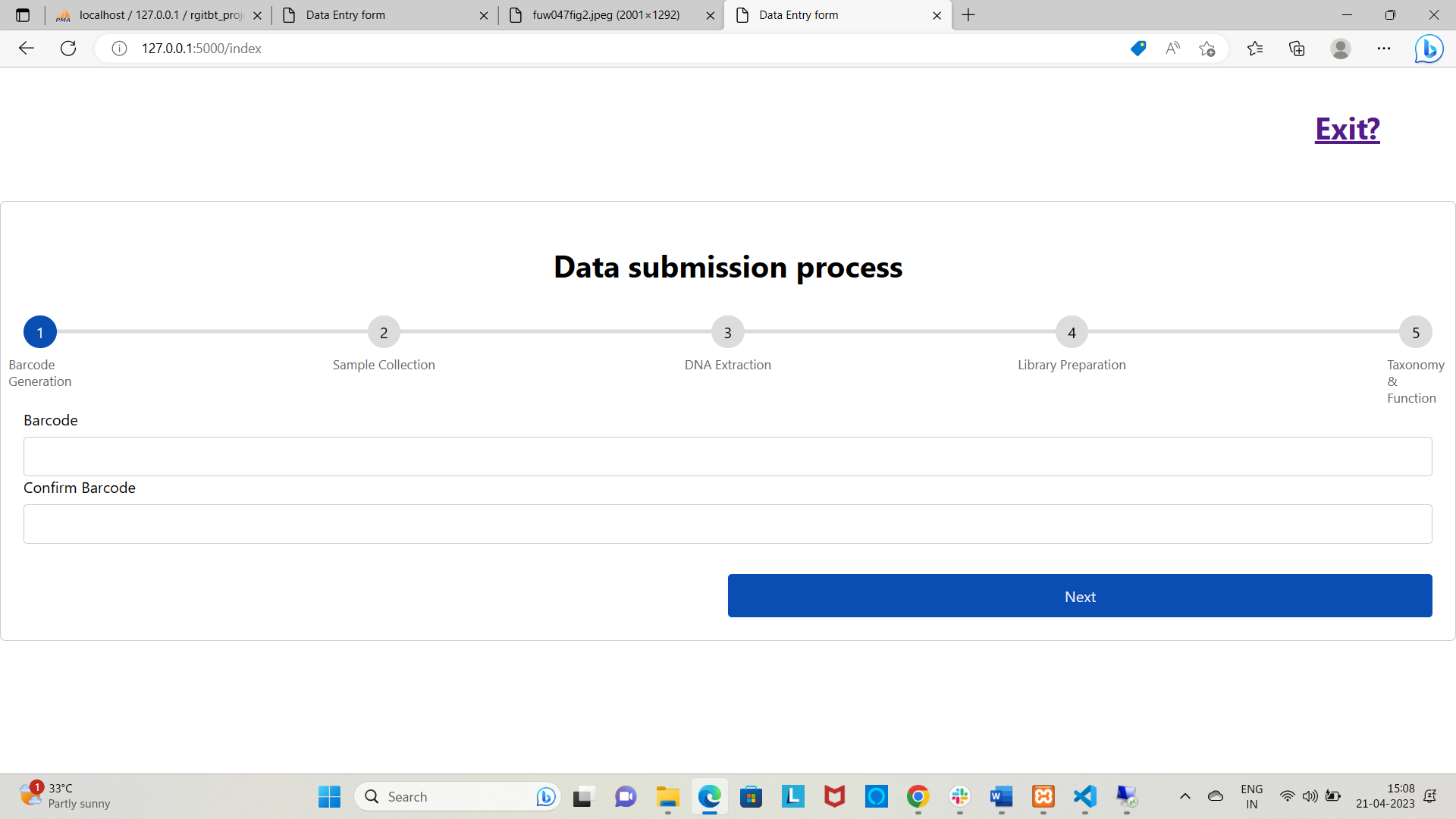
AJAX stands for Asynchronous JavaScript and XML. In a nutshell, it is the use of the XMLHttpRequest object to communicate with servers. It can send and receive information in various formats, including JSON, XML, HTML, and text files.

**2.4.2 Pseudo code:**

Data submission portal contains five sections:

1. **Barcode Generation:**

A barcode is generated from where each sample will be identified as unique. This barcode will be transfer for each table. Once this barcode is created it will appear in each section in the data submission form.



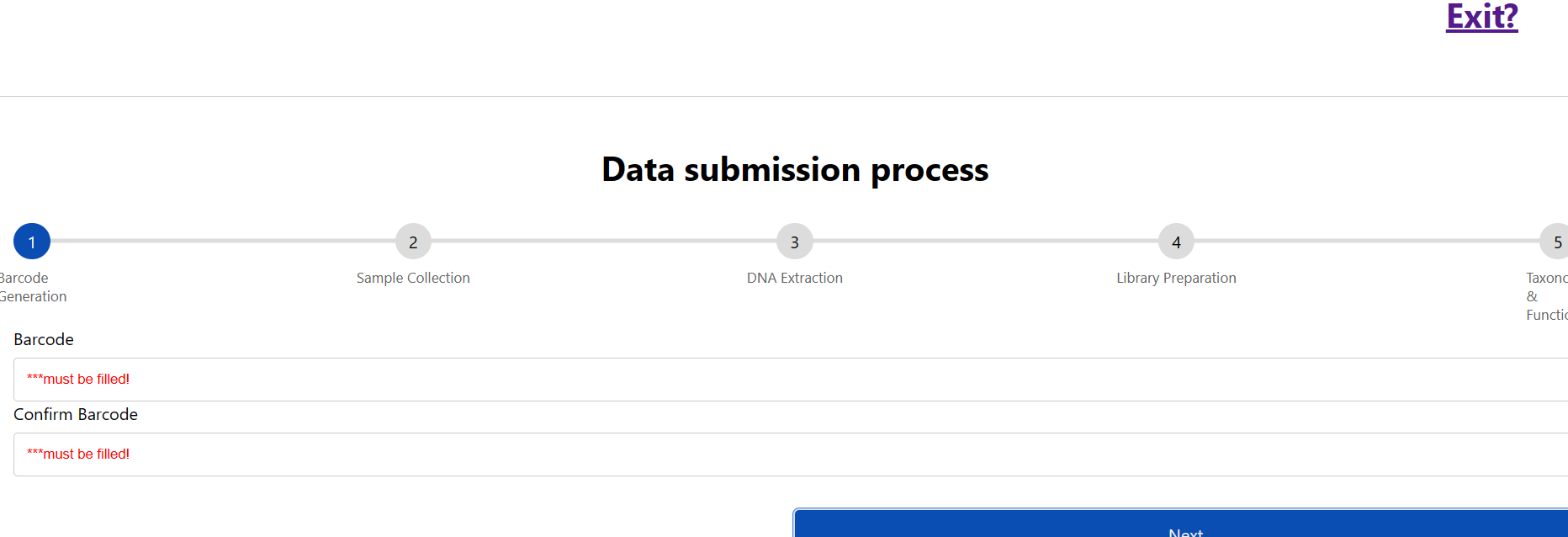
After filling barcode and then clicking next button it will bring to the next section of the data submission process. We can terminate here by clicking Exit button. Barcode will be generated.

Data from the front-end to the flask-end will be sent using Jquery JavaScript ajax function. For each section there is a single JavaScript dedicated to send the data.

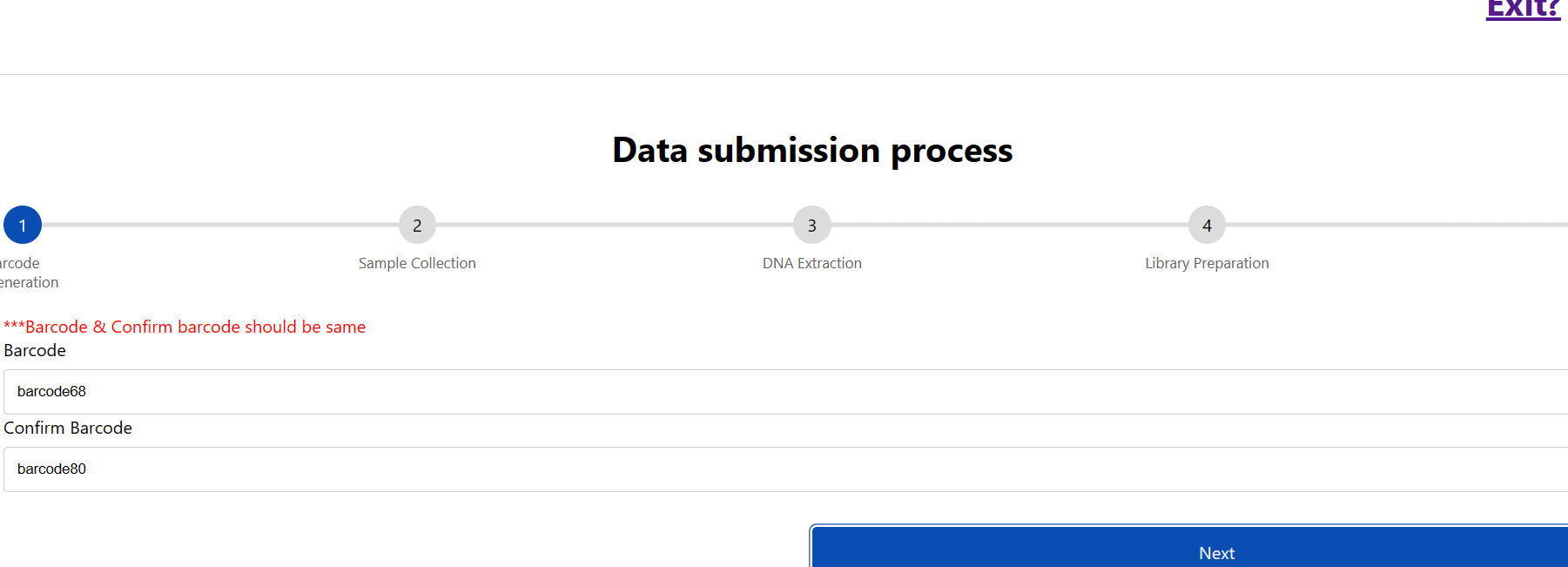
For Barcode generation *form1.js*, for Sample collection *form2.js*, for DNA Extraction *form3.js*, for library preparation *form4.js* and for taxonomy file upload *upload.js* are made to send its specific data to the flask

**Form validation feature: -**

A validation is added for empty field. If user tries to go to the next section without entering any values it will give warning to fill the fields first.

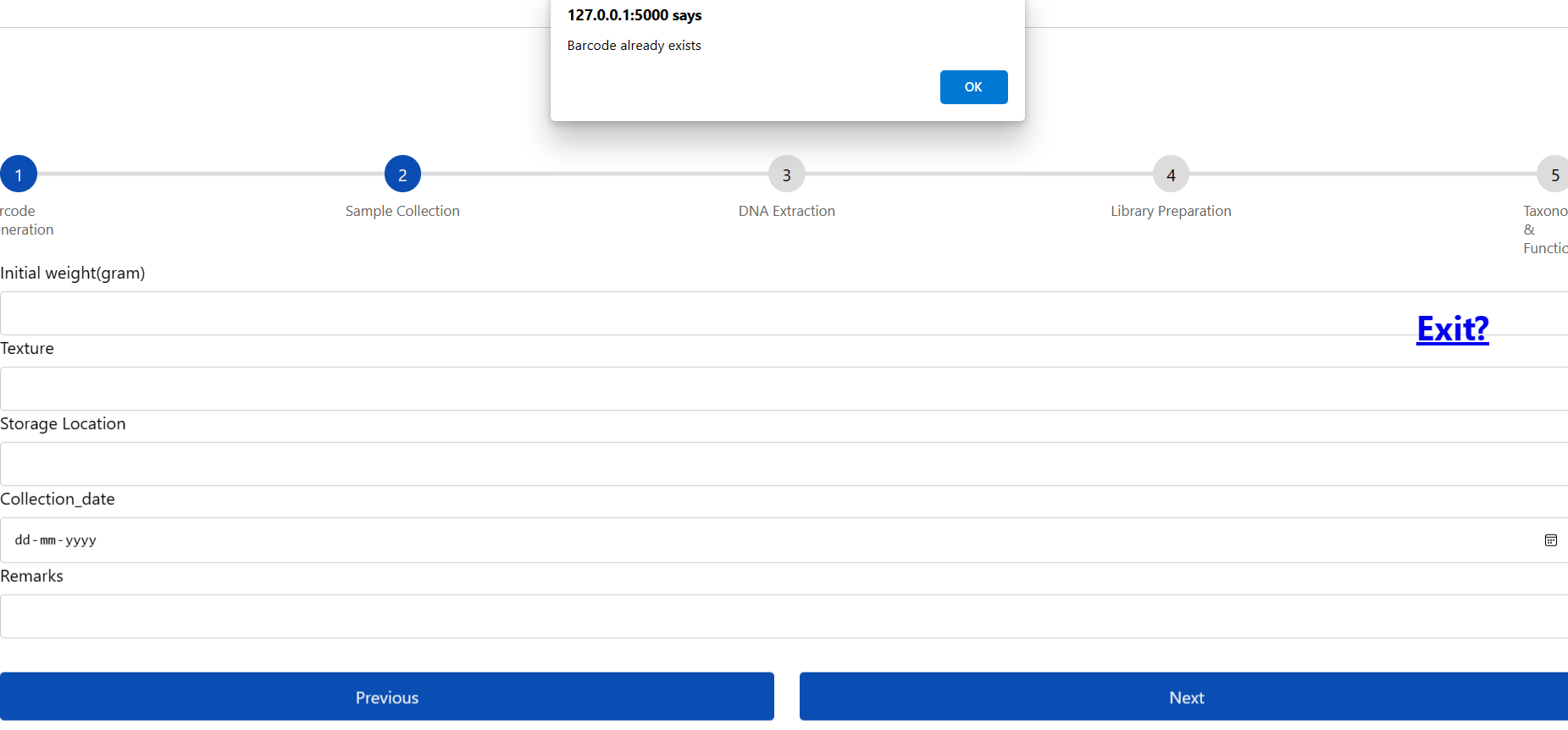


If barcode and confirm barcode does not match, it gives a warning. Adding this feature makes user more confident as this barcode will be tracking in each step.



**Preventing duplicate entry of a barcode:**

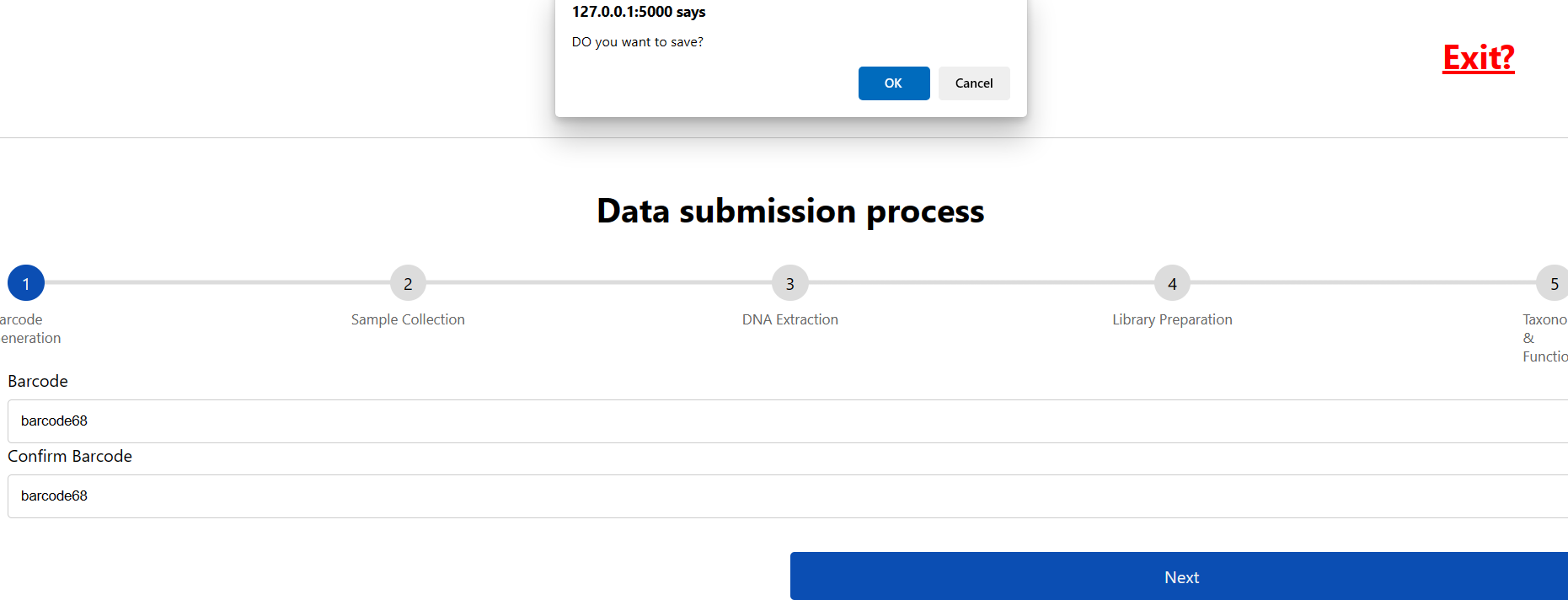
For duplicate entry it will show an alert popup.



In the database, barcode is kept as primary key. Since barcode is considered as sample ID. So for each sample unique ID there should be Unique for barcode also. For this reason barcode is considered as primary key. If user enters any pre-existing barcode, that can be prevented. After appearing duplicate entry popup, it will redirect the page into the first section of data submission again.

**Exit from the process: -**

If user wants to terminate the process, just clicking exit button in the right corner of the page, will bring the user back to home page. There will be a dialogue box which will give a confirmation massage for saving data. If user wants to choose *cancel* option, there will be no termination in the process.



Once all these validations are succeeded, one can proceed to the next.

1. **Sample collection:**

Fields for this section are

1. Initial weight
2. Texture
3. Storage
4. Collection Date
5. Remarks

*Initial weight* is the weight of stool sample that are collected from customer. Unit for this weight is always in gram.

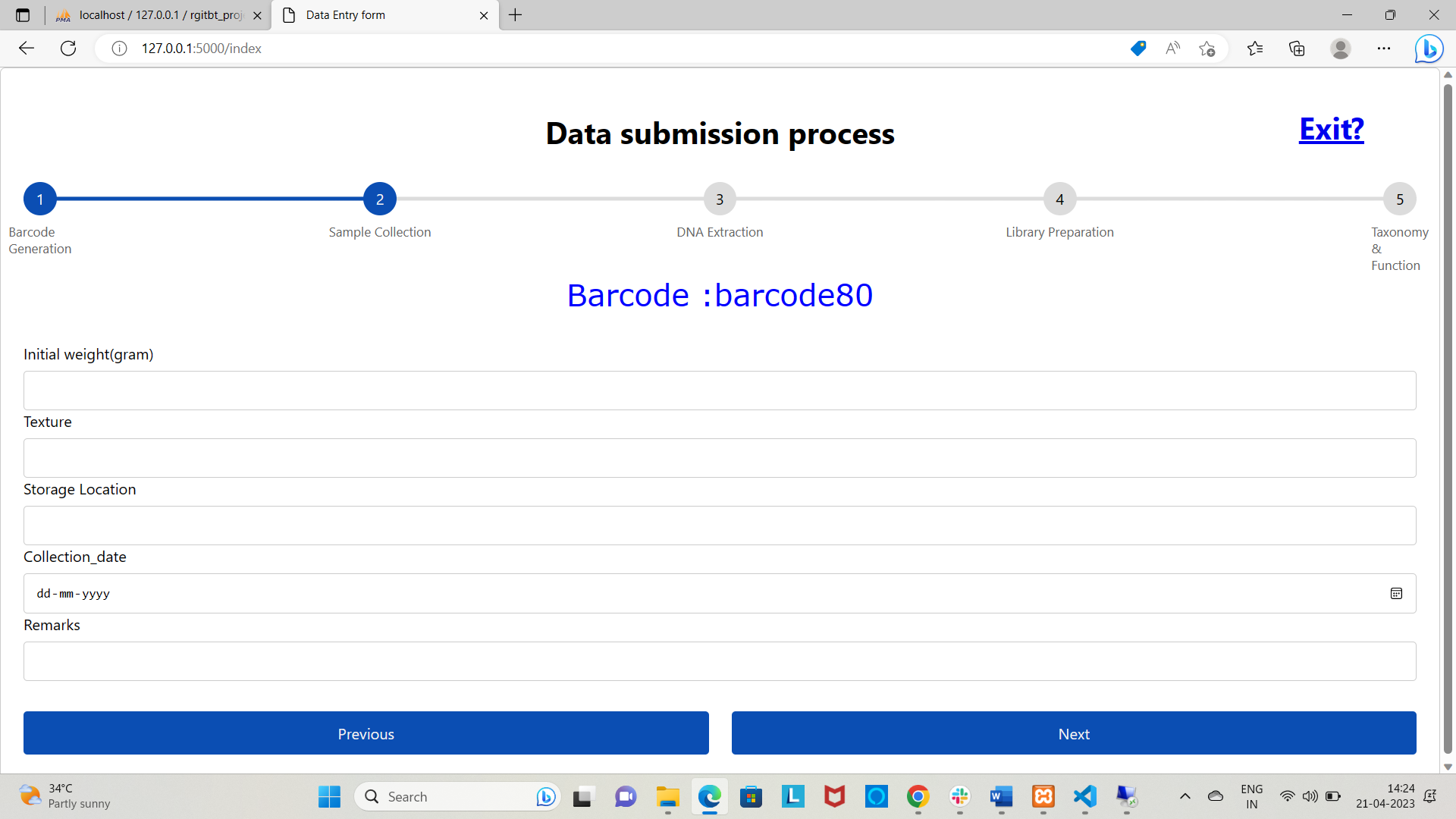
*Texture* tells about the condition of the sample. Default is smooth.

*Storage* is the location of the sample. Generally, it is the place and the condition where the sample is kept. Default is refrigerator at the temperature of -10 degree Celsius.

*Collection date* is the date when sample is brought to the laboratory.

*Remarks* tells whether Sample is qualified or not for nanopore run.

**Entry fields for sample collection: -**

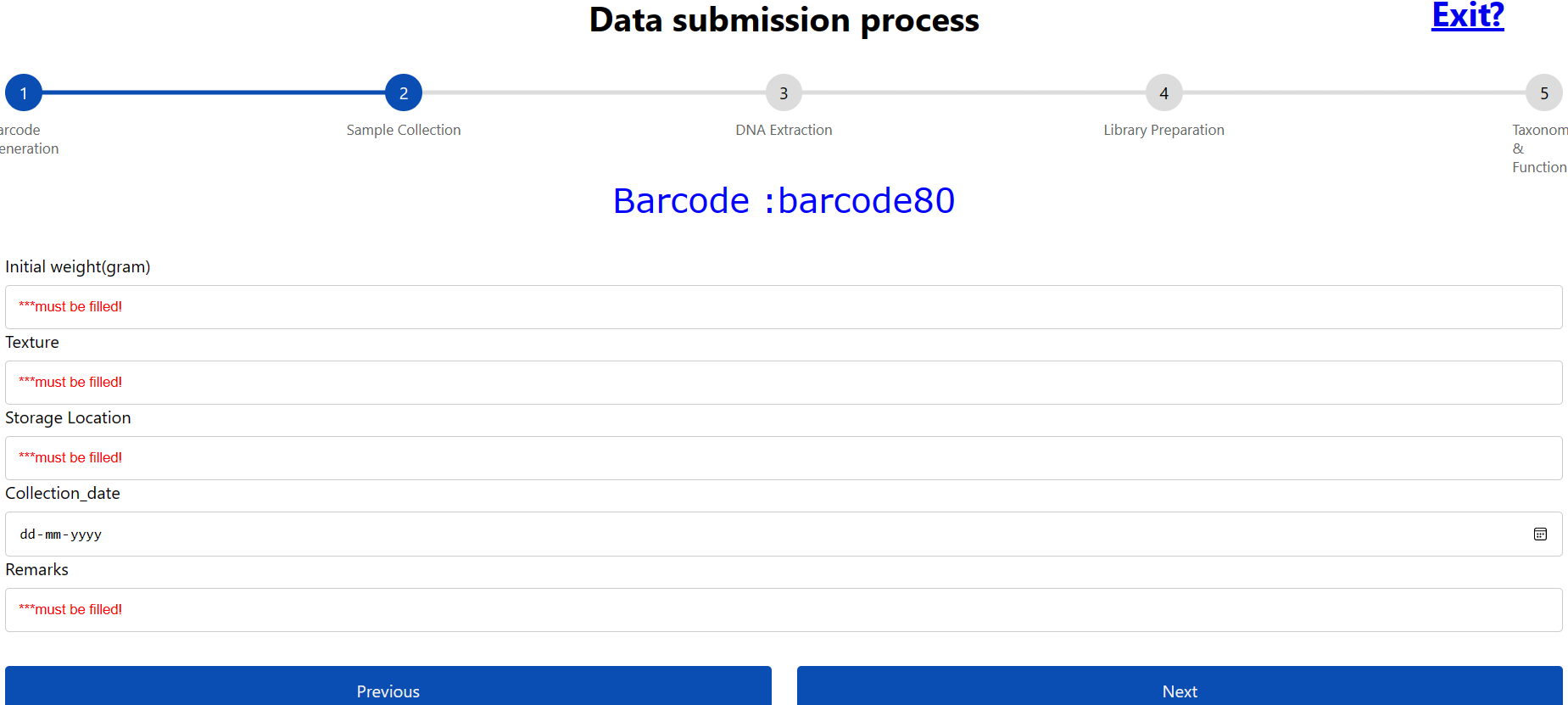


Since in the first section (Barcode Generation) barcode80 as for demo is entered which is displayed in the top. This barcode acts as a primary key for saving this all-fields related sample collection. This flow of *barcode display* is made in such a way that every time user enters a barcode in the barcode section, will appear rest section in the submitting form.

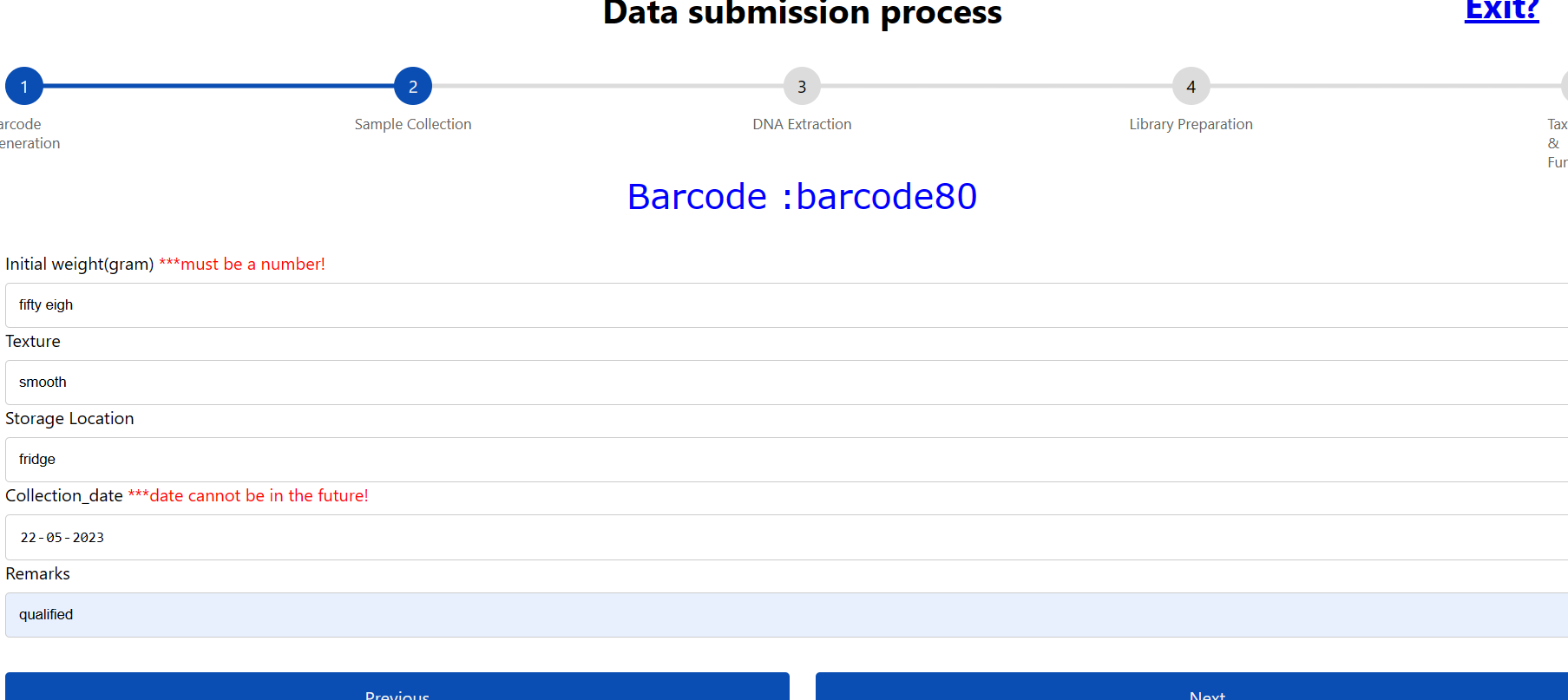
After clicking Next button down below, it will bring the next section in the form. And if we click on exit button in the above, we can terminate the process but data from this section will be saved as mentioned in the previous section.

**Form validation features: -**

This section of code is for validation of empty field, it will give a warning massage for each field.



This section is for validating alpha numeric value and for date field that should not be in future date otherwise it will raise warning.



After following all these above-mentioned validations, user can further proceed to the next section of data submission.

1. **DNA Extraction:**

Fields for DNA Extraction are:

1. Kit name
2. Lot no
3. Quantity
4. Qubit conc.
5. Nanodrop Conc.
6. Nanodrop 260/280
7. Nanodrop 260/230
8. Extraction date
9. Repeat no.
10. Remarks

**Kit name:** There are many different DNA extraction kits available on the market, each with their own specific protocol and reagents. Some commonly used DNA extraction kits are Qiagen fast stool mini kit, Qiagen Powerfaecal kit .

**Lot no:** A lot number (or batch number) is a unique identifier given to a specific batch of reagents or consumables used during the extraction process. The lot number helps to track the origin and quality of the materials used, and it can be useful in case of any troubleshooting or quality control issues. The lot number is typically printed on the packaging or label of the reagents or consumables, and it can be recorded in the documentation or database for the sequencing project.

**Quantity:** DNA extraction quantity. Unit is always milligram

**Qubit concentration:** Qubit is a fluorometric assay used to measure the concentration of DNA in a sample. The Qubit assay is more accurate and sensitive than the traditional UV spectrophotometry method. Quantity is measured as mg/ml.

**Nanodrop concentration:** In the Nanodrop assay, a small sample volume (typically 1-2 µL) is placed on a sample pedestal and a beam of UV-Vis light is passed through the sample. The amount of light absorbed by the sample is then measured to determine the concentration of nucleic acids or proteins. Quantity is measured in mg/ml

**Nanodrop 260/280:** The Nanodrop 260/280 ratio is a measure of the purity of nucleic acid samples (DNA or RNA) based on the absorbance readings at two different wavelengths: 260 nm and 280 nm.

The absorbance at 260 nm is primarily due to the presence of nucleic acids, while the absorbance at 280 nm is due to the presence of both nucleic acids and proteins. Therefore, the ratio of the absorbance at 260 nm to 280 nm can indicate the relative amounts of nucleic acids and proteins in the sample.

**Nanodrop 260/230:** The Nanodrop 260/230 ratio is another measure of the purity of nucleic acid samples (DNA or RNA) based on the absorbance readings at two different wavelengths: 260 nm and 230 nm.

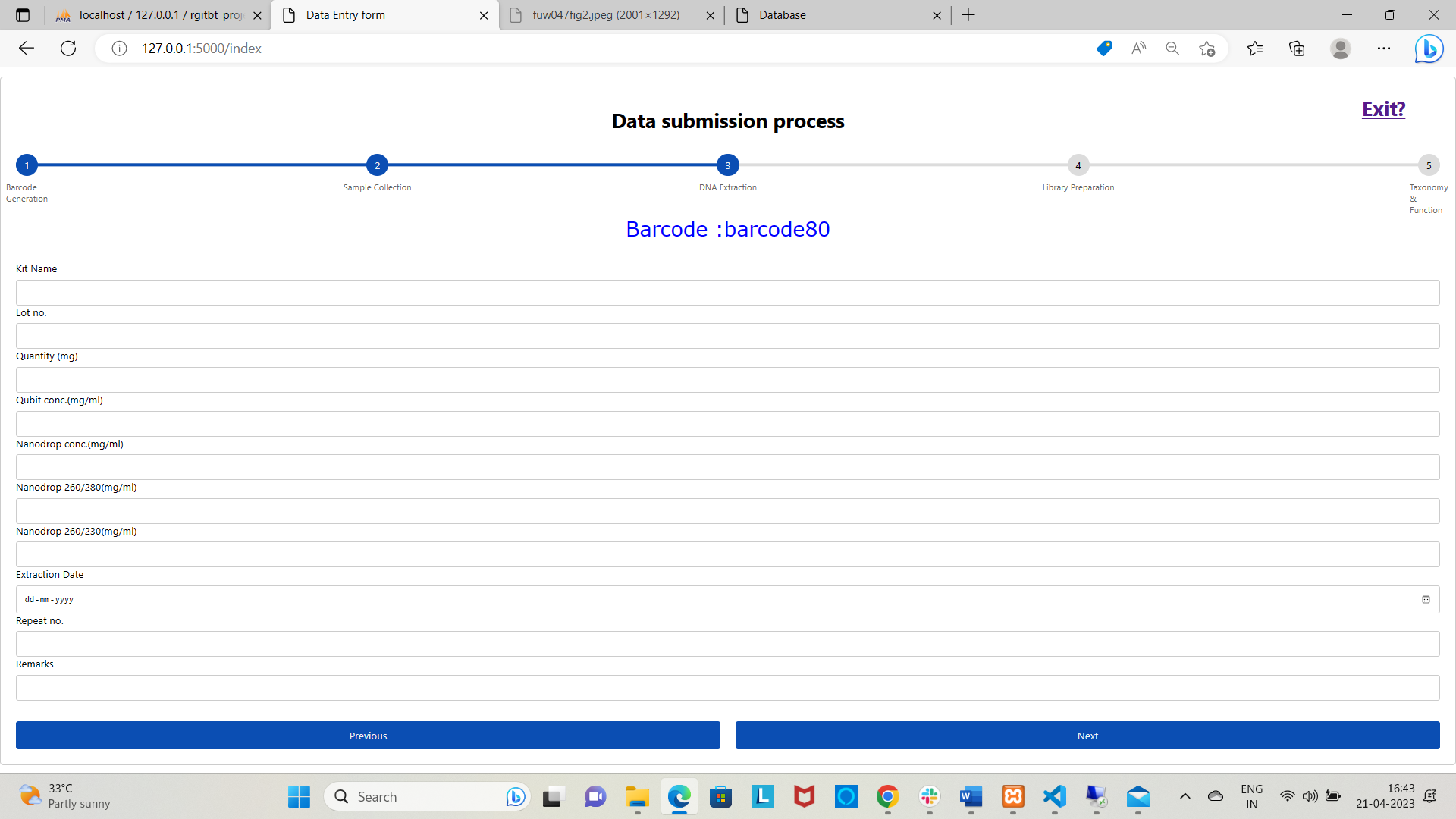
The absorbance at 260 nm is primarily due to the presence of nucleic acids, while the absorbance at 230 nm is due to the presence of contaminants such as phenol, guanidine, or other organic compounds that can interfere with downstream applications. Therefore, the ratio of the absorbance at 260 nm to 230 nm can indicate the relative amounts of nucleic acids and contaminants in the sample.

**Extraction Date:** Date when the extraction of DNA process takes place.

**Repeat no.:** For any failed dna Extraction repeat no will track the process. Default is zero

**Remarks:**  Qualified or not. For repetition of the process there should be a reason to be mentioned for failing the experiment.

**Field entry for DNA Extraction: -**



**Form validation features: -**

This section is for validation of empty field. It will throw a warning for each mandatory field.

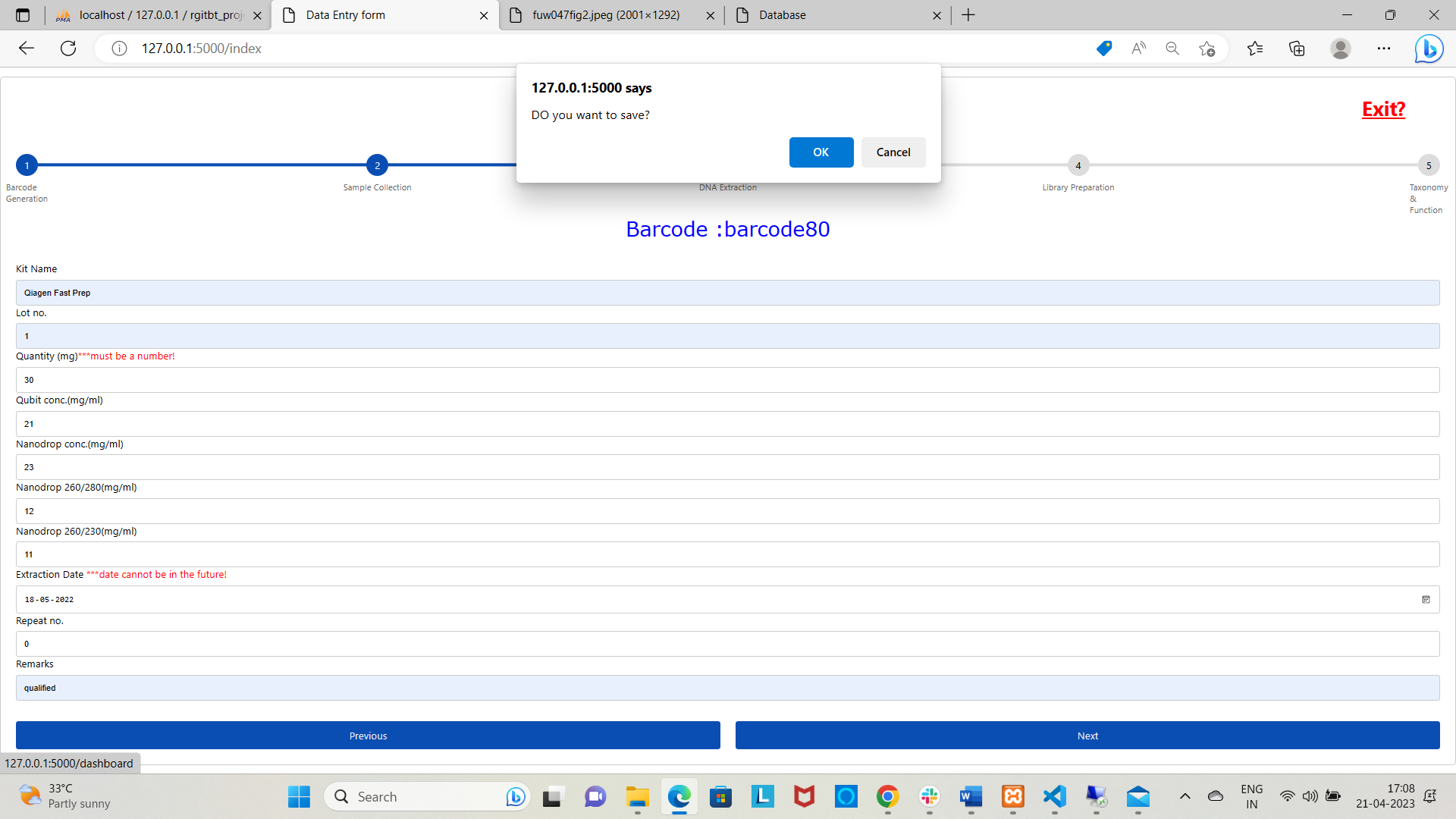


This section is for validating numeric validation and also date validation for any future date.



**Exit from the process: -**

This section is for terminating the workflow using exit button.



1. **Library Preparation:**

Field for this section:

1. Run ID
2. Extraction start date
3. Extraction End date
4. Yield nanodrop
5. Yield qubit
6. Library Start date
7. Library end date
8. Flow cell ID
9. Native Barcode
10. Active pores
11. Pores remaining
12. Loading start
13. Loading end
14. Sequence status
15. Sequence duration
16. Status

**Run ID**: A Nanopore Run ID is a unique identifier given to a single sequencing run on a Nanopore sequencer. It is generated by the instrument software during the sequencing run and is used to track the data generated from that run throughout the analysis pipeline.

**Extraction start date and end date:** Extraction start date and extraction end date refer to the dates when DNA or RNA extraction was initiated and completed, respectively.

**Yield qubit and Yield nanodrop:** Output from nanodrop concentration and qubit concentration.

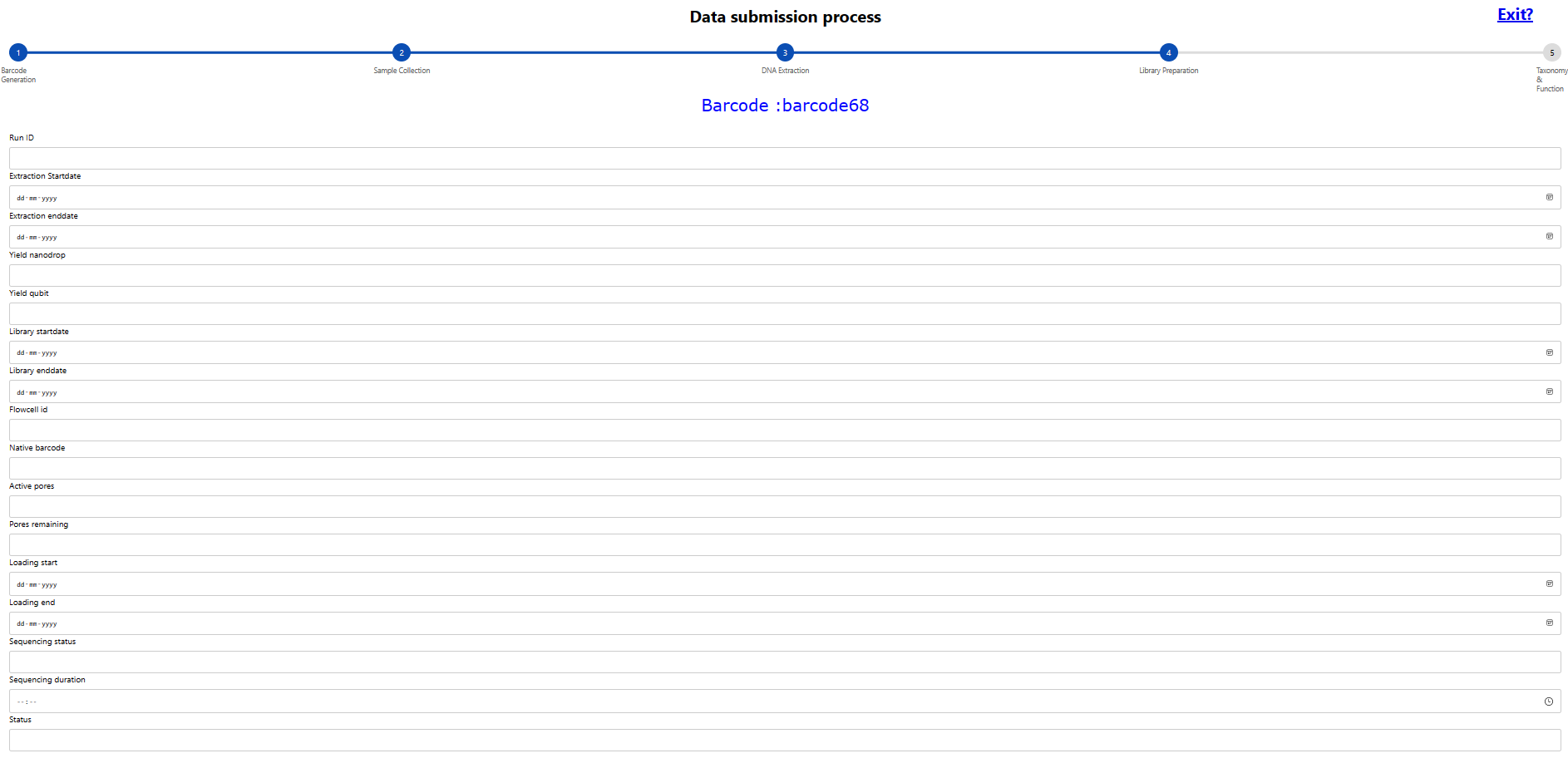
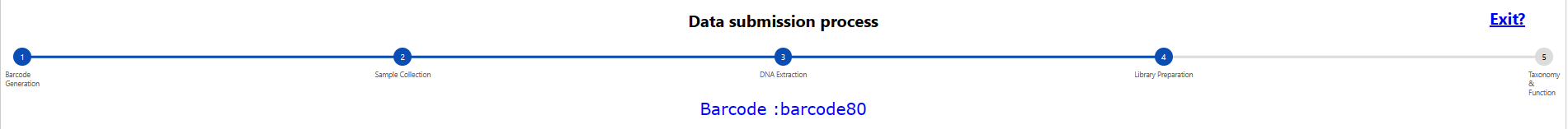
**Library start date and Library end date:** Library start date and library end date refer to the dates when the library preparation process for sequencing was initiated and completed, respectively.

**Flow cell ID:** A flow cell ID is a unique identifier given to a specific flow cell used in a Nanopore sequencing run.

**Native Barcode:** In Nanopore sequencing, native barcodes are short DNA sequences that are added to the genomic DNA or cDNA samples prior to library preparation. These barcodes allow for the multiplexing of multiple samples in a single sequencing run, which can increase sequencing efficiency and reduce the cost per sample.

**Loading start date and end date:** Loading start date and end date refer to the dates when the samples were loaded onto the sequencing instrument and the loading process was completed, respectively.

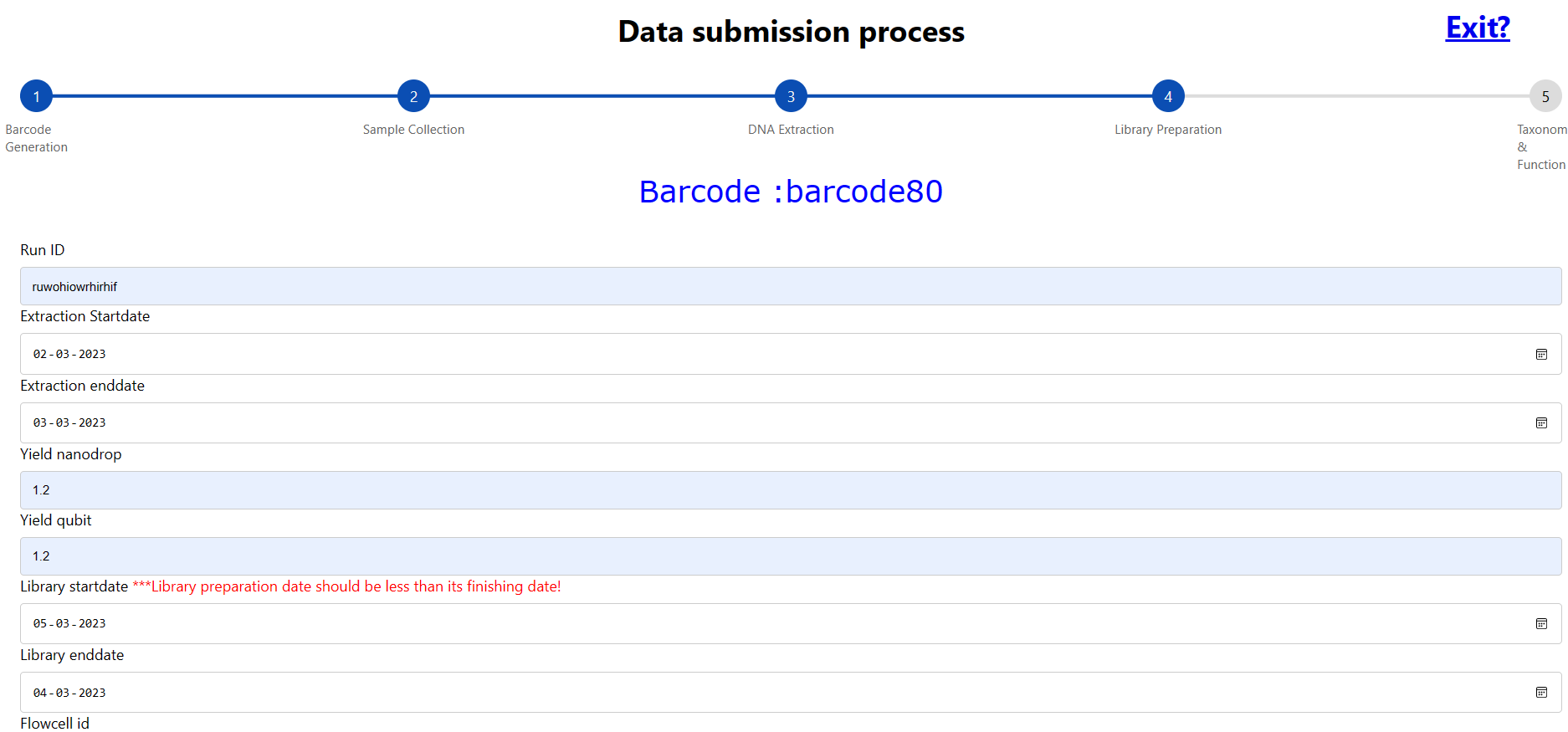
**Entry fields for library preparation: -**



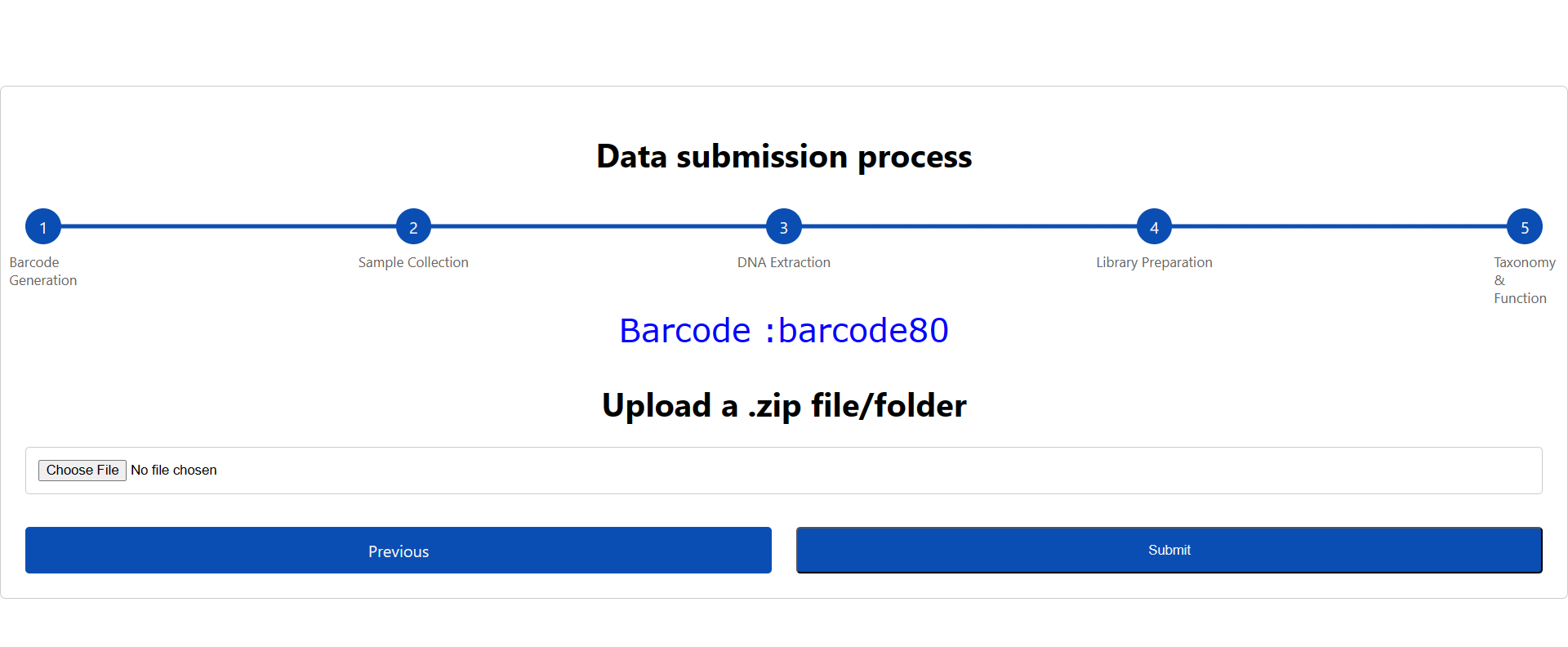


**Form validation: -**

Since there are different field having different dates, those date value should not collide for this trying to attempt a validation. E.g. (a warning will come if library preparation end date is past than its start date). Here a snippet has been added: -

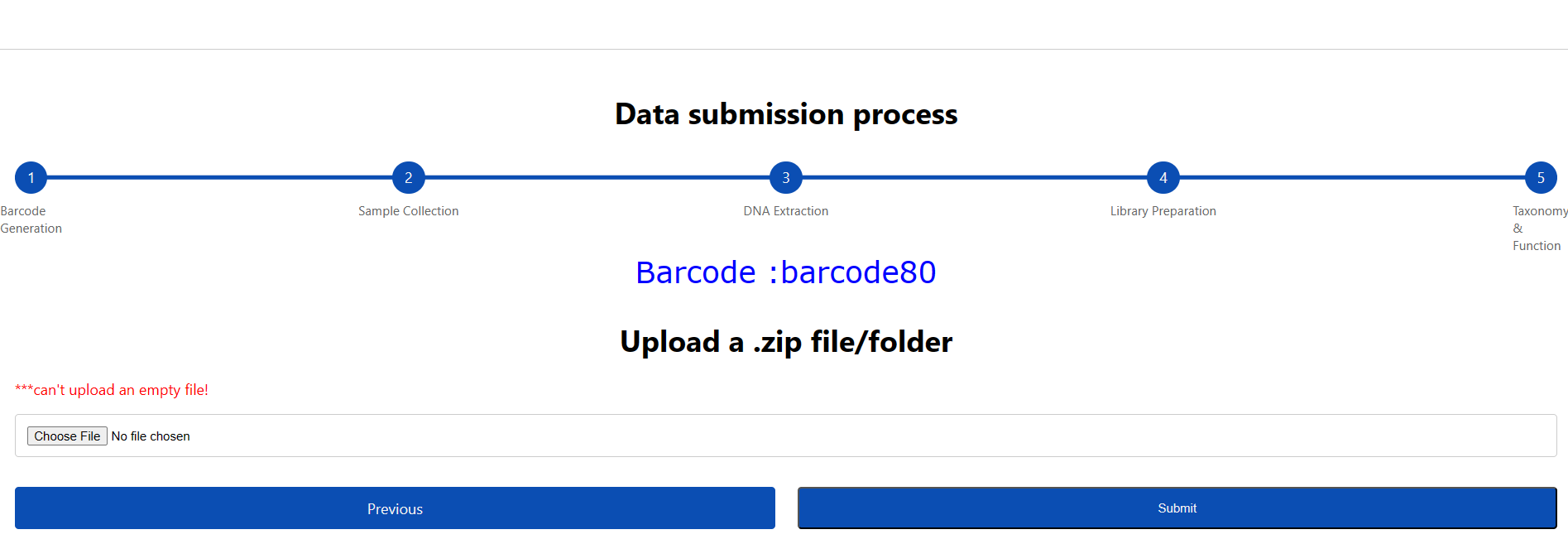


1. **Taxonomy File upload:**



**Form validation: -**

This validation will prevent in submitting without uploading file. There will be an alert for this.

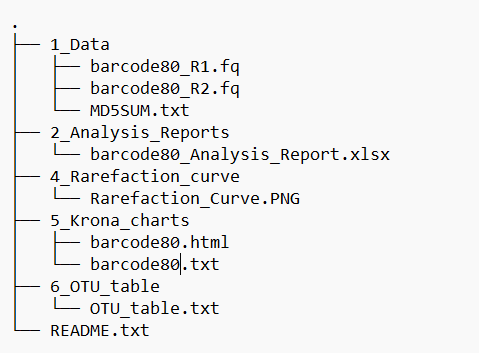


This is for processing the .zip file for uploading.

**2.4.3 Condition for uploading file:**

Submitting portal only takes .zip file format. For any taxonomic analysis, it is always expected to get multiple files and directory. From there it is not possible to search a single file and upload from there which will be big problem for user. To avoid such clashes, a script has been set , which will automatically read all the files from the directory and will take which one is necessary.

**2.4.3.1 Directory folder for zip format:**



from this directory “Analysis\_Report.xlsx” is the targeting file. In this mentioned file, all taxonomic bacterial profile is mentioned with their relative abundance. For each level of taxonomy (e.g., Kingdom, Phylum, Order, Class, Genus, Species) separate sheet is provided. From their only species sheet is considered for uploading data.

**2.4.4 Methods for uploading files:**

We can make all the content from the zip folder in a list then extract that file using substring match with the help of python module.

**2.5 Output:**

*16S taxonomy file analysis* refers to the process of analyzing the output files generated from a 16S rRNA gene sequencing experiment to identify and classify the microbial taxa present in a sample.

This system is made for uploading a folder contains all output from a taxonomic analysis in the form of zip file. From there it will extract a file where it contains bacterial taxonomic abundance at species level.

In this file it will show all bacterial name present in a sample along with its abundance. Bacterial profile will be mentioned like Kingdom, Phylum, Order, Class, Genus, Species.

With the help of total species count from the column of *Abundance count* (which basically counts species present in a sample for each species) alpha diversity indices are calculated (Shannon, Simson and Pielou score) and all value will be stored in the backend database.

**Shannon Score:**

Shannon diversity index is a measure of alpha diversity, which describes the diversity of species within a single community or ecosystem. It is a widely used metric in ecology and is often used to quantify the diversity of microbial communities in metagenomic sequencing studies.

The Shannon diversity index, also known as Shannon entropy or Shannon-Wiener index, takes into account both the number of species (richness) and the evenness of their distribution. It is calculated as follows:

H' = -Σ(pi \* log2(pi))

where pi is the proportion of individuals belonging to the ith species, and log2 is the logarithm with base 2. The index ranges from 0 (no diversity) to a maximum value that depends on the number of species present in the community.

**Simson score:**

The Simpson diversity index is a measure of alpha diversity, which describes the diversity of species within a single community or ecosystem. It is commonly used in ecology to quantify the diversity of plant and animal communities, and in metagenomic sequencing studies to quantify the diversity of microbial communities.

The Simpson diversity index is calculated as follows:

D = Σ(ni / N)^2

where ni is the number of individuals belonging to the ith species, N is the total number of individuals in the community, and Σ is the sum over all species.

**Pielou score:**

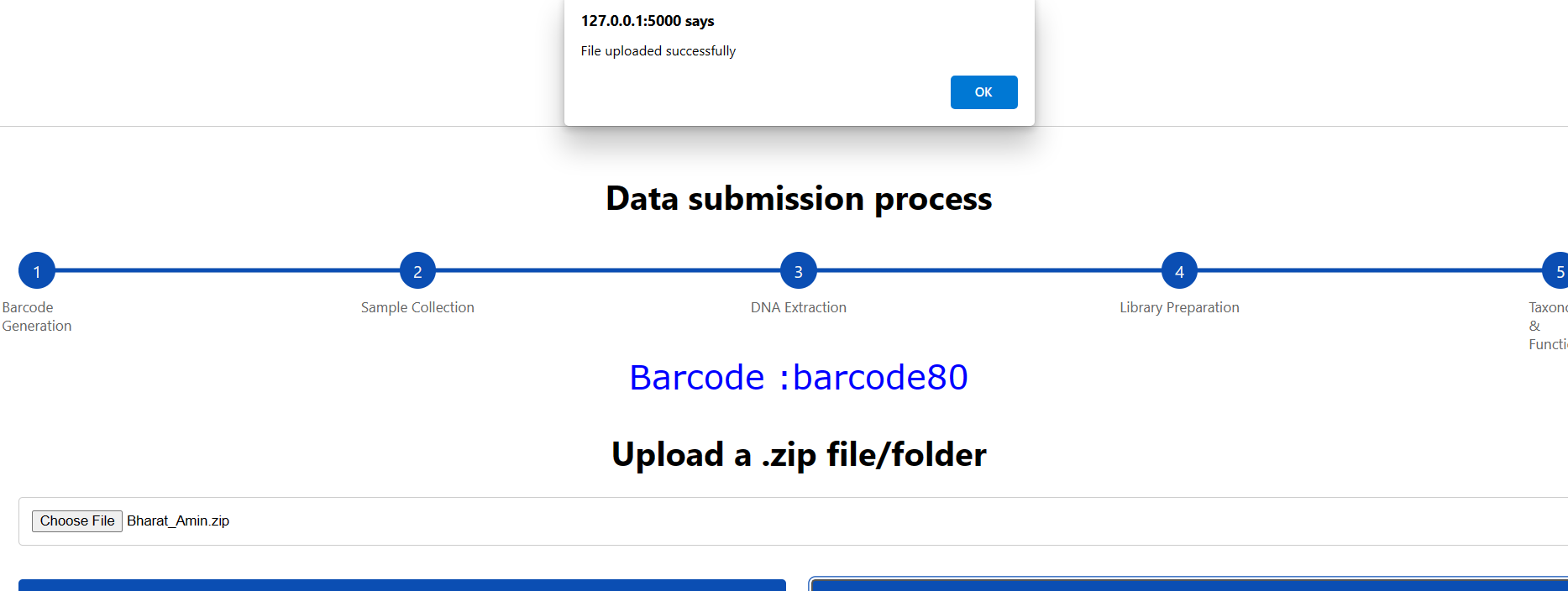
The Pielou's evenness index, also known as the J' index, is a measure of evenness that describes how equally the abundance of individuals is distributed among the different species present in a community or ecosystem. It is commonly used in ecology and is often used in combination with other diversity indices, such as the Shannon diversity index or the Simpson diversity index, to provide a more comprehensive description of the diversity and richness of communities.

The Pielou's evenness index is calculated as follows:

J' = H' / ln(S)

where H' is the Shannon diversity index, as described in my previous answer, and S is the total number of species present in the community.

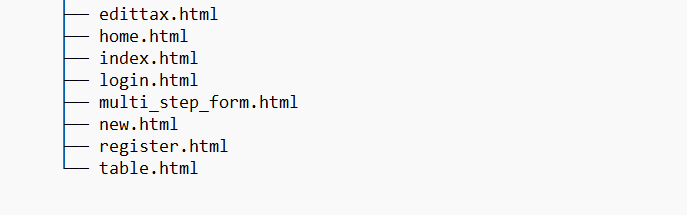
After uploading file one can proceed with submission the data.



Once this submission success massage displayed, it will redirect to the dashboard.

**2.6 Flask integration with front-end and back-end: -**

For running a script from flask, it requires two folder and one main script file. In this project how the flask directory will look like is shown below: -



As we can see there is a main.py file and two folders static and templates folder. Static folder is a directory for storing static files such as images,CSS files, Javascript files and other assets. This directory is generally restricted for client. Templates folder is a directory for storing all HTML files. Each HTML file is rendered through main.py file using render\_template function from flask module.

**2.7 main.py: -**

This file is main application file of my project. From where I defined my flask application, database configuration and defined the route for the application. The first section of the file contains all the necessary module for this project. And second section is meant for sql-database configuration. Then its different route for different html pages which are rendered from templates folder in its place.

**CHAPTER 3**

**RESULTS AND DISCUSSION**

**3.1 Results: -**

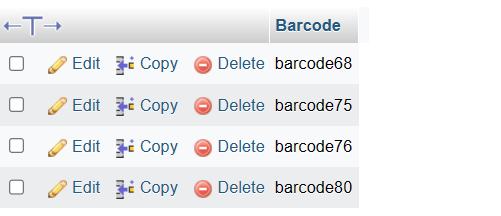
With the above mention methodology, the software is developed with the consequent result.

For each step submission, Data will be stored in the back-end mysql database

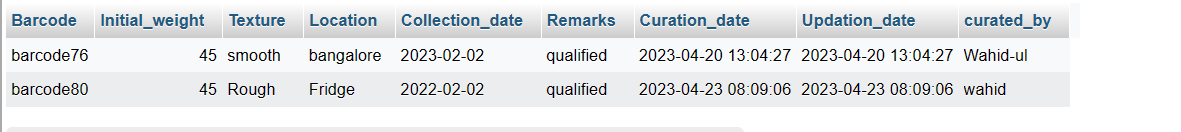
From xampp server we can see how data will be stored.

Lab data:

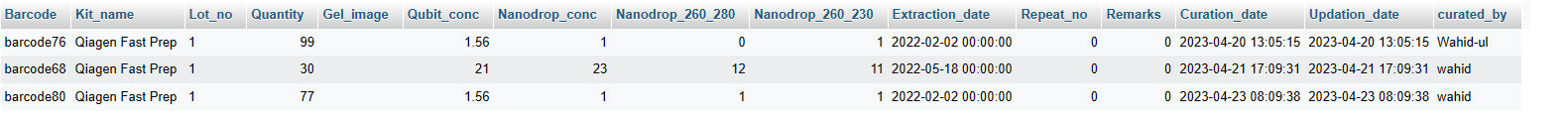
**Barcode metadata:**

****

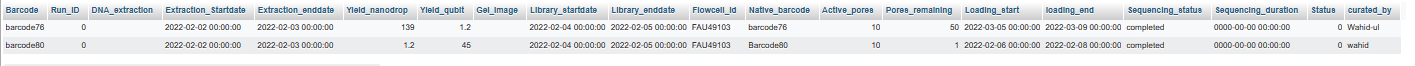
**Sample collection:**



**DNA Extraction data:**

**e**

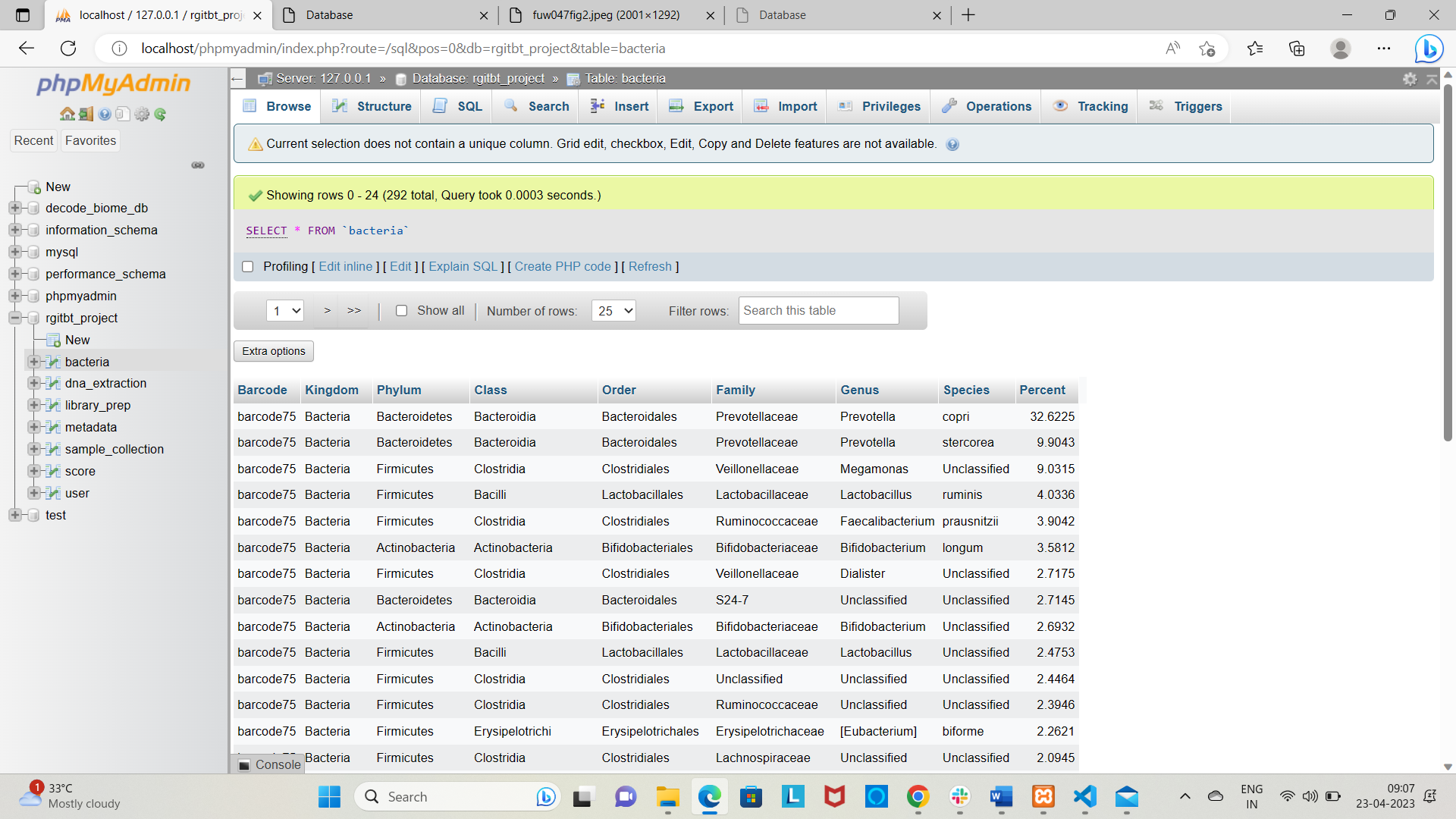
**Library Preparation data:**

****

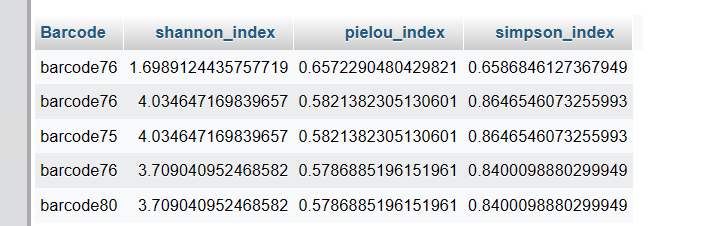
***Note: - Here curation date, updation-date and curated by (generally user log in name) are added in every lab table in the back-end which is hidden from front-end client end. I don’t want to show this thing in front part as it is not relevant for a client but for tracking data efficiently it is a crucial step to collect all these information, all these things will be processed internally.***

**Taxonomy file data:**

Bacterial data:

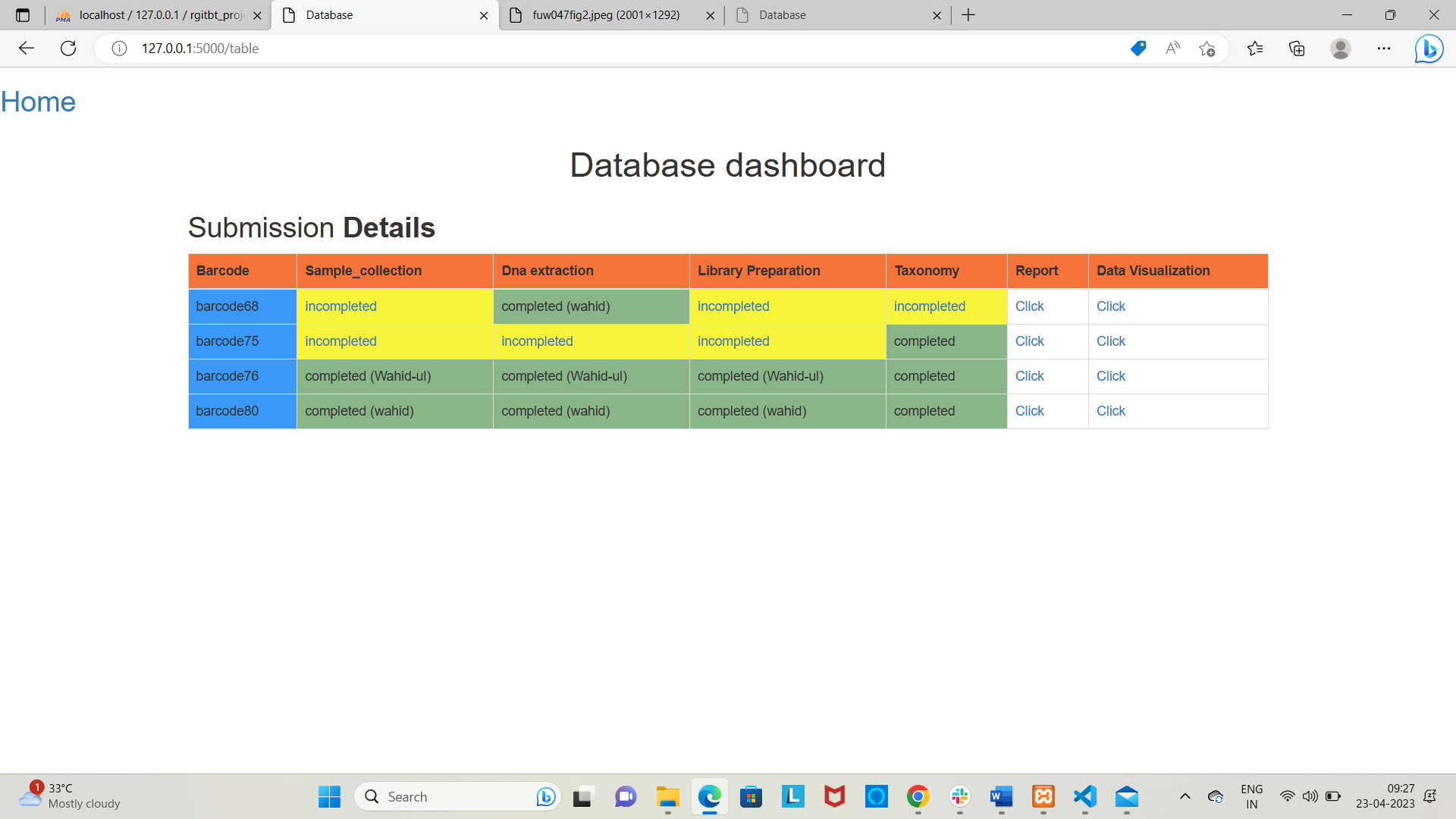


Score table:

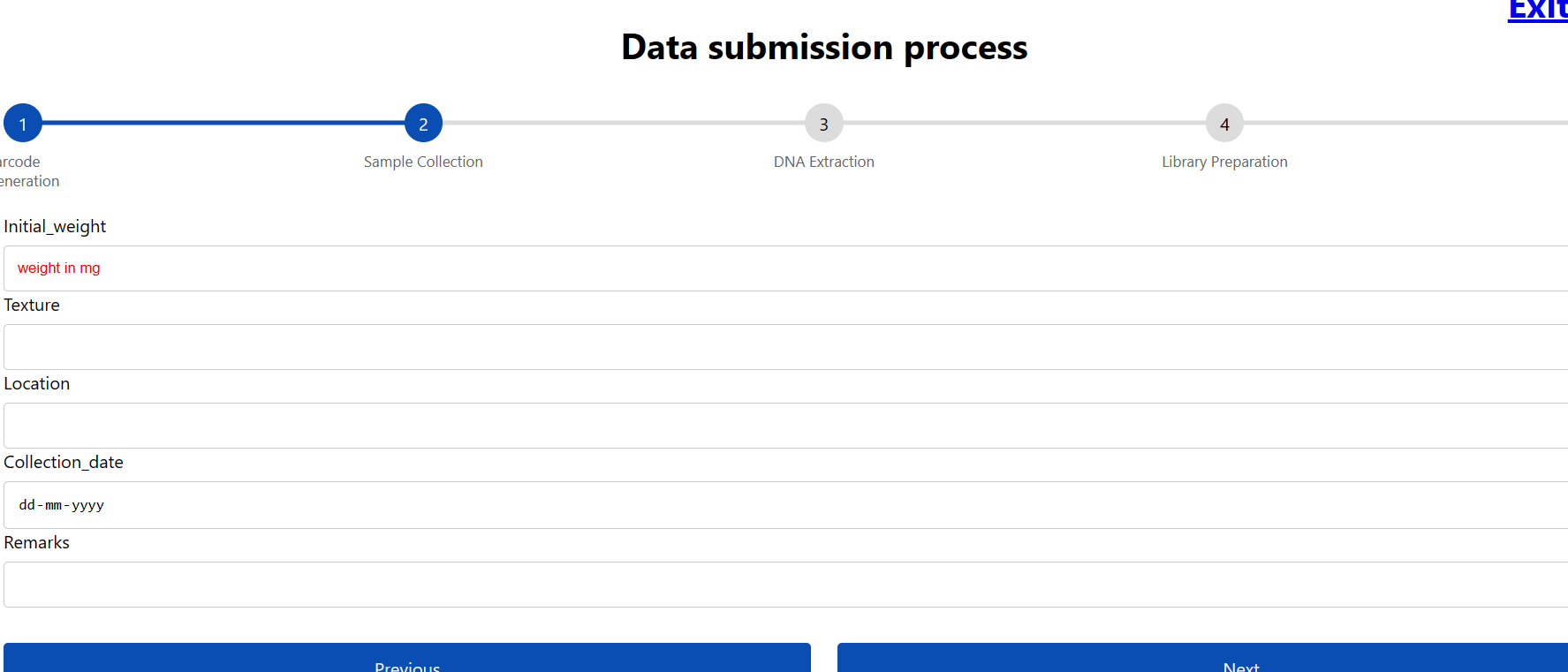


**3.2 Added Feature:**

These above-mentioned data stored in the back-end are restricted to the client-side. Since back end is the part of storing data only. To make them visualize for client I have added all tables in a report file. These can be fetched directly from the back-end using flask.



From this dashboard, one can see what is the status of each section of data. For example, for barcode68 we can see in sample collection, it is showing incomplete status, after clicking this link it will redirect to that section only. Just like this: -



Same formula is applied for other section also.

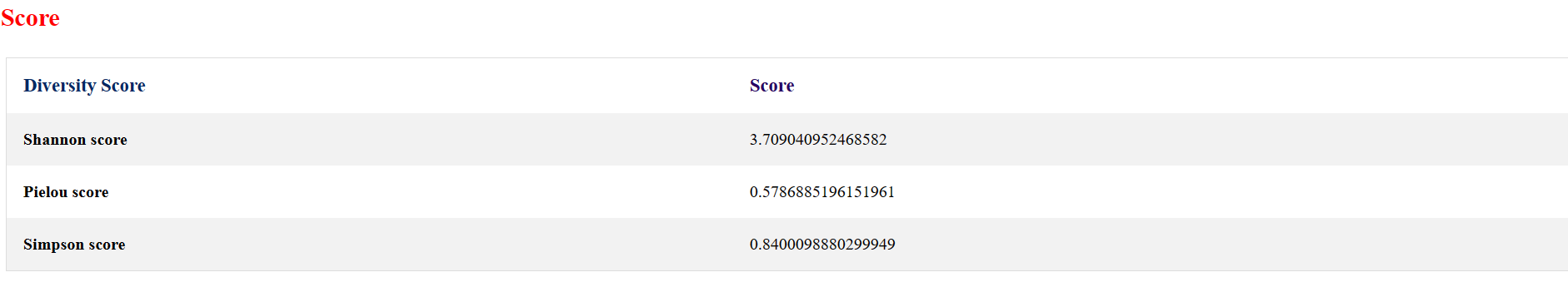
In the report section from the header, user can see what are the information is stored relevant to that barcode.

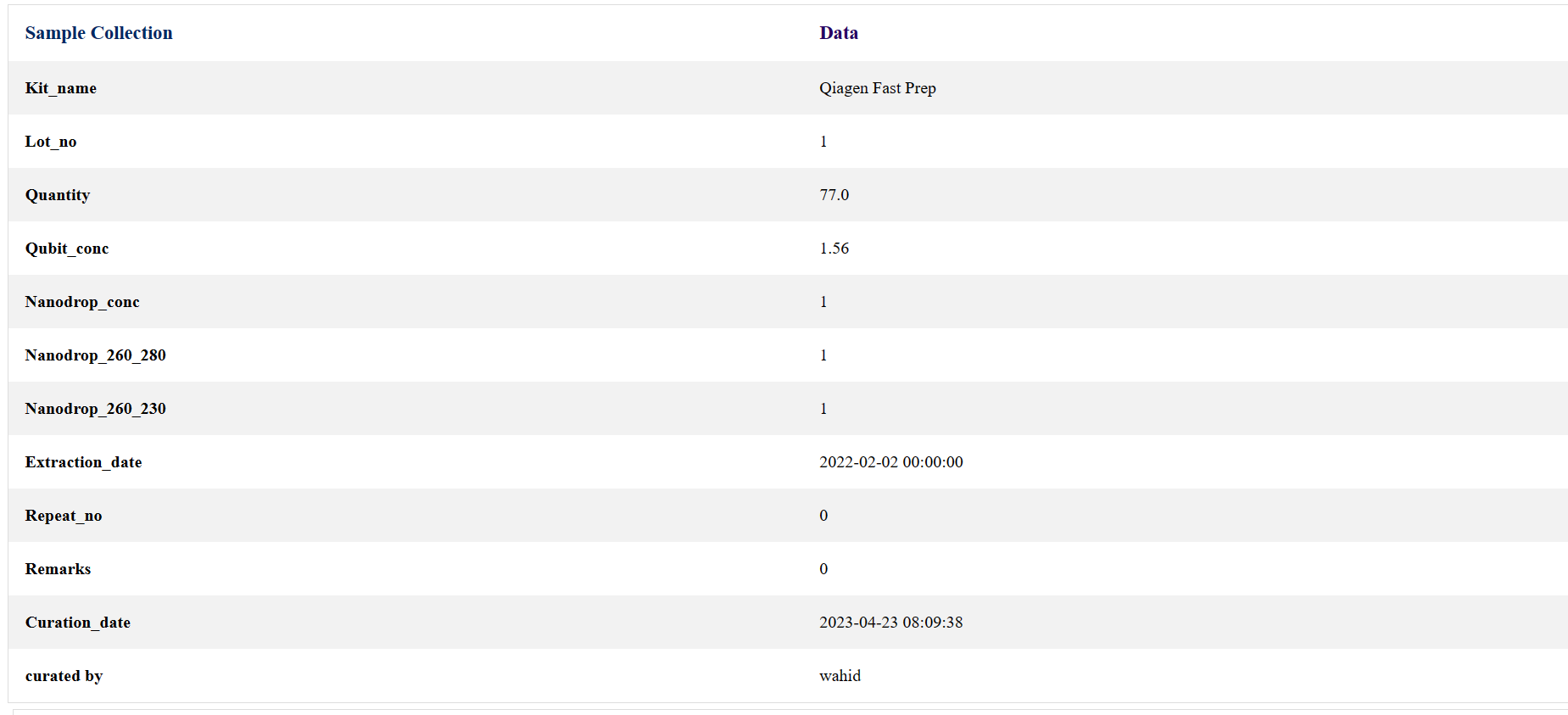


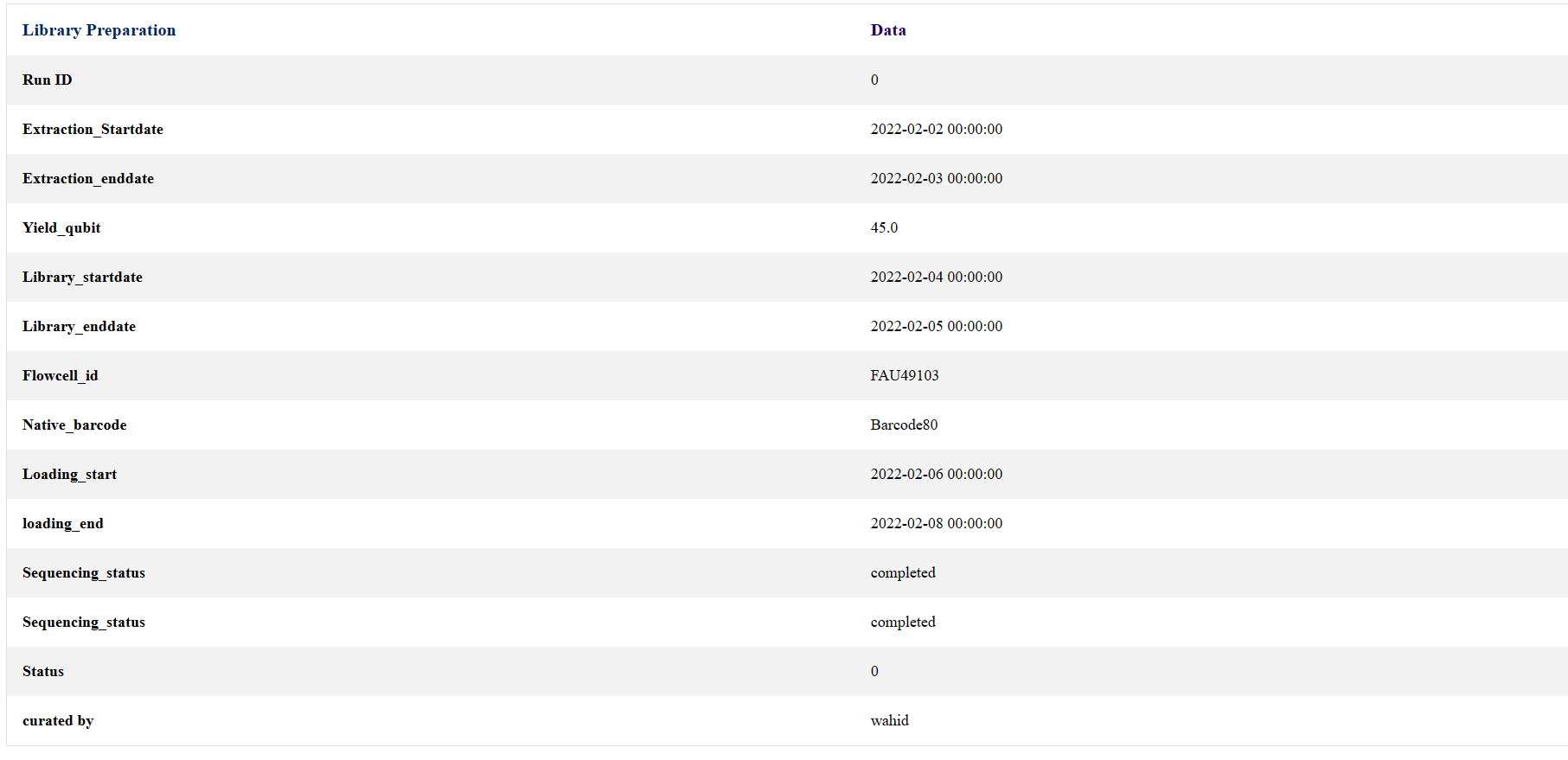
Since for barcode68 we only have dna extraction data, rest are empty so for this we can only see dana extraction table and other remains as empty.

Now for barcode76 and barcode80 we can see all sections are completed so report will look like: -

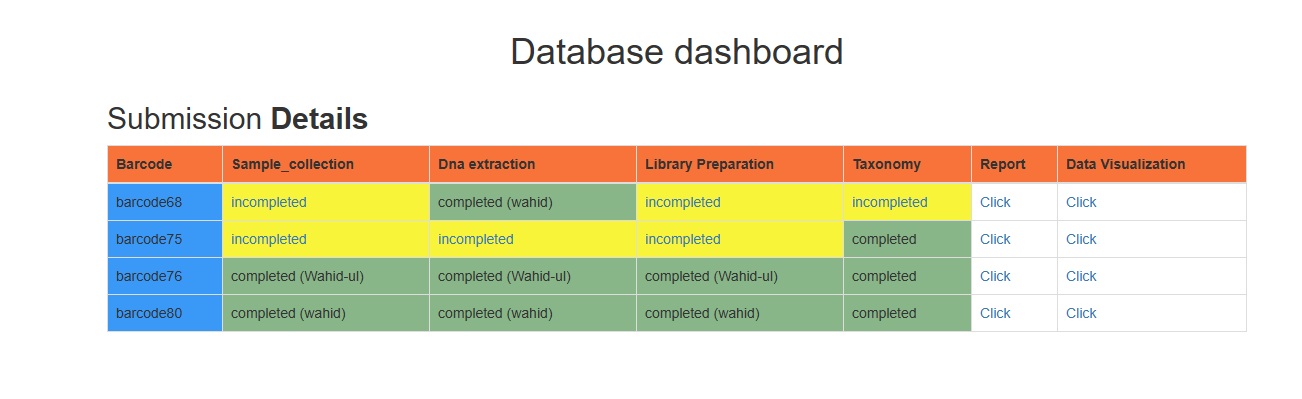




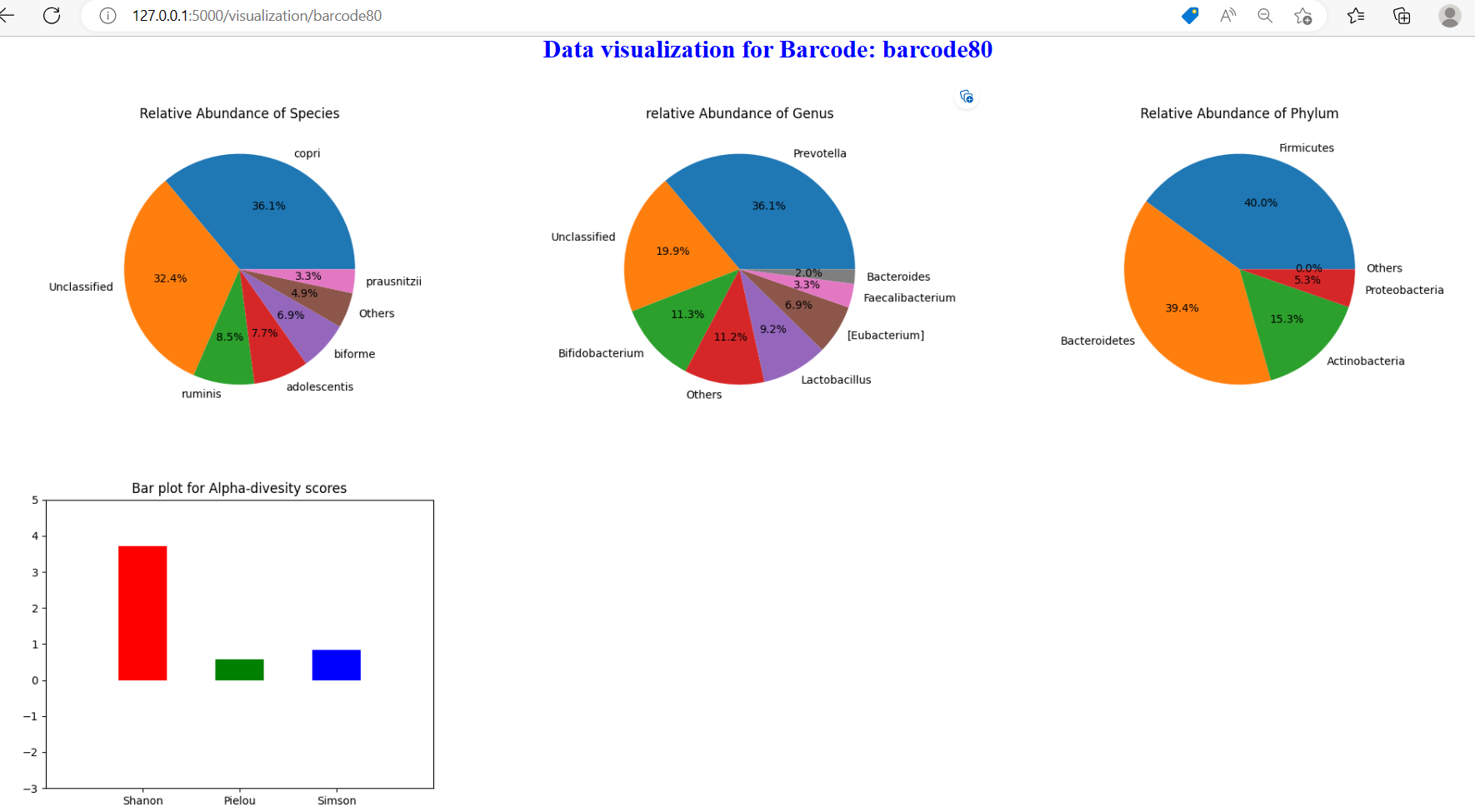
 



For making better analysis, features have been added for visualizing various graphical interface:



For the dashboard, data visualization section contains pie chart of relative abundance of different level (Species, Genus, Phylum). Clicking any specific barcode, it will give some amazing plots. In this demo code if I click barcode80, we can generate plots that are shown below: -



**3.3 Discussion: -**

The aim of this project was to develop a software which will allow the users to perform various operations under a single roof. The main objective is management of data starting from wet-lab protocol up to taxonomic analysis. Lab protocol, size, quantity and contamination factors play an important role on taxonomic analysis.

The problems one can face for managing huge metagenomics data: -

1. Unstructured data.
2. There is minimum connection between lab-protocol and taxonomic analysis.
3. Any changes from lab-protocol can change taxonomic interpretation.

From this software we can give a frame to unstructured data as we are collecting all data and tracking by a single barcode, now data is not unstructured any more.

**CHAPTER 4**

**CONCLUSION**

**4.1 Conclusion:**

The desired API software is developed not only for lab data management also to view some outcomes of 16S taxonomic analysis.

The basic aim of the software was fulfilled as it allows the user to find a significant difference in the abundance of phylum, genus, species level. Also, it helps to calculate various alpha diversity indices. Besides this software is also providing a link between lab processing for sequencing. It provides all interpretation in the graphical view which makes users to understand better in visualizing the outputs.

**CHAPTER 6**

**BIBLIOGRAPHY**

**5.1 Bibliography: -**

1. Conrad Bessant, Darren Oakley, Ian Shadforth: Building Bioinformatics Solutions with perl,R and SQL
2. ProFacGen. (n.d.). Bioinformatics Data Management. Retrieved April 25, 2023, from <https://www.profacgen.com/Bioinformatics-Data-Management.htm> .
3. David Broyles,..Brian E. Dixon, in Health information Exchange,2016 <https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/database-management-system#:~:text=A%20database%20management%20system%20(DBMS,you%20create%20and%20maintain%20databases>.
4. Torsten Thomas, Jack Gilbert and Folker Meyer on February 9,2012 in National Library of Medicine: Metagenomics- a guide from sampling to data analysis. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3351745/#:~:text=Metagenomics%20is%20defined%20as%20the,DNA%20%5B2%2C3%5D>.
5. Published in December 30,2015 on nature methods <https://www.nature.com/articles/nmeth.3718> .
6. Liz Simmons: Front-end vs Back-end: What’s the difference? <https://www.computerscience.org/bootcamps/resources/frontend-vs-backend/#:~:text=What's%20the%20Difference%20Between%20Front,system%2C%20data%2C%20and%20logic>.
7. Handelsman J, Rondon MR, Brady SF, Clardy J, Goodman RM. Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products. Chem Biol. 1998;5(10):R245–249. doi: 10.1016/S1074-5521(98)90108-9.
8. Tyson GW, Chapman J, Hugenholtz P, Allen EE, Ram RJ, Richardson PM, Solovyev VV, Rubin EM, Rokhsar DS, Banfield JF. Community structure and metabolism through reconstruction of microbial genomes from the environment. Nature. 2004;428(6978):37–43. doi: 10.1038/nature02340. [PubMed] [CrossRef] [Google Scholar]
9. Venter JC, Remington K, Heidelberg JF, Halpern AL, Rusch D, Eisen JA, Wu D, Paulsen I, Nelson KE, Nelson W, Fouts DE, Levy S, Knap AH, Lomas MW, Nealson K, White O, Peterson J, Hoffman J, Parsons R, Baden-Tillson H, Pfannkoch C, Rogers YH, Smith HO. Environmental genome shotgun sequencing of the Sargasso Sea. Science. 2004;304(5667):66–74. doi: 10.1126/science.1093857. [PubMed] [CrossRef] [Google Scholar]
10. Simon C, Daniel R. Metagenomic analyses: past and future trends. Appl Environ Microbiol. 2011;77(4):1153–1161. doi: 10.1128/AEM.02345-10. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
11. Wilmes P, Bond PL. Metaproteomics: studying functional gene expression in microbial ecosystems. Trends Microbiol. 2006;14(2):92–97. doi: 10.1016/j.tim.2005.12.006. [PubMed] [CrossRef] [Google Scholar]
12. Gilbert JA, Field D, Huang Y, Edwards R, Li W, Gilna P, Joint I. Detection of large numbers of novel sequences in the metatranscriptomes of complex marine microbial communities. PLoS One. 2008;3(8):e3042. doi: 10.1371/journal.pone.0003042. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
13. Beja O, Aravind L, Koonin EV, Suzuki MT, Hadd A, Nguyen LP, Jovanovich SB, Gates CM, Feldman RA, Spudich JL, Spudich EN, DeLong EF. Bacterial rhodopsin: evidence for a new type of phototrophy in the sea. Science. 2000;289(5486):1902–1906. doi: 10.1126/science.289.5486.1902. [PubMed] [CrossRef] [Google Scholar]
14. Nicol GW, Schleper C. Ammonia-oxidising Crenarchaeota: important players in the nitrogen cycle? Trends Microbiol. 2006;14(5):207–212. doi: 10.1016/j.tim.2006.03.004. [PubMed] [CrossRef] [Google Scholar]
15. Burke C, Kjelleberg S, Thomas T. Selective extraction of bacterial DNA from the surfaces of macroalgae. Appl Environ Microbiol. 2009;75(1):252–256. doi: 10.1128/AEM.01630-08. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
16. Delmont TO, Robe P, Clark I, Simonet P, Vogel TM. Metagenomic comparison of direct and indirect soil DNA extraction approaches. J Microbiol Methods. 2011;86(3):397–400. doi: 10.1016/j.mimet.2011.06.013. [PubMed] [CrossRef] [Google Scholar]
17. Knight R, Desai N, Field D, Fierer N, Fuhrman J, Gordon J, Hu B, Hugenholtz P, Jansson J, Meyer F, Stevens R, Bailey M, Kowalchuk G, Gilbert J. Designing Better Metagenomic Surveys: The role of experimental design and metadata capture in making useful metagenomic datasets for ecology and biotechnology. Nature Biotechnology. in review.
18. Thomas T, Rusch D, DeMaere MZ, Yung PY, Lewis M, Halpern A, Heidelberg KB, Egan S, Steinberg PD, Kjelleberg S. Functional genomic signatures of sponge bacteria reveal unique and shared features of symbiosis. ISME J. 2010;4(12):1557–1567. doi: 10.1038/ismej.2010.74. [PubMed] [CrossRef] [Google Scholar]
19. Palenik B, Ren Q, Tai V, Paulsen IT. Coastal Synechococcus metagenome reveals major roles for horizontal gene transfer and plasmids in population diversity. Environ Microbiol. 2009;11(2):349–359. doi: 10.1111/j.1462-2920.2008.01772.x. [PubMed] [CrossRef] [Google Scholar]
20. Angly FE, Felts B, Breitbart M, Salamon P, Edwards RA, Carlson C, Chan AM, Haynes M, Kelley S, Liu H, Mahaffy JM, Mueller JE, Nulton J, Olson R, Parsons R, Rayhawk S, Suttle CA, Rohwer F. The marine viromes of four oceanic regions. PLoS Biol. 2006;4(11):e368. doi: 10.1371/journal.pbio.0040368. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
21. Abbai NS, Govender A, Shaik R, Pillay B. Pyrosequence analysis of unamplified and whole genome amplified DNA from hydrocarbon-contaminated groundwater. Mol Biotechnol. 2011. [PubMed]
22. Lasken RS. Genomic DNA amplification by the multiple displacement amplification (MDA) method. Biochem Soc Trans. 2009;37(Pt 2):450–453. [PubMed] [Google Scholar]
23. Ishoey T, Woyke T, Stepanauskas R, Novotny M, Lasken RS. Genomic sequencing of single microbial cells from environmental samples. Curr Opin Microbiol. 2008;11(3):198–204. doi: 10.1016/j.mib.2008.05.006. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
24. Goltsman DS, Denef VJ, Singer SW, VerBerkmoes NC, Lefsrud M, Mueller RS, Dick GJ, Sun CL, Wheeler KE, Zemla A, Baker BJ, Hauser L, Land M, Shah MB, Thelen MP, Hettich RL, Banfield JF. Community genomic and proteomic analyses of chemoautotrophic iron-oxidizing "Leptospirillum rubarum" (Group II) and " Leptospirillum ferrodiazotrophum" (Group III) bacteria in acid mine drainage biofilms. Appl Environ Microbiol. 2009;75(13):4599–4615. doi: 10.1128/AEM.02943-08. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
25. Sorek R, Zhu Y, Creevey CJ, Francino MP, Bork P, Rubin EM. Genome-wide experimental determination of barriers to horizontal gene transfer. Science. 2007;318(5855):1449–1452. doi: 10.1126/science.1147112. [PubMed] [CrossRef] [Google Scholar]
26. Metzker ML. Sequencing technologies - the next generation. Nat Rev Genet. 2010;11(1):31–46. doi: 10.1038/nrg2626. [PubMed] [CrossRef] [Google Scholar]
27. Mardis ER. The impact of next-generation sequencing technology on genetics. Trends Genet. 2008;24(3):133–141. doi: 10.1016/j.tig.2007.12.007. [PubMed] [CrossRef] [Google Scholar]
28. Niu B, Fu L, Sun S, Li W. Artificial and natural duplicates in pyrosequencing reads of metagenomic data. BMC Bioinformatics. 2010;11:187. doi: 10.1186/1471-2105-11-187. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
29. Teal TK, Schmidt TM. Identifying and removing artificial replicates from 454 pyrosequencing data. Cold Spring Harb Protoc. 2010;2010(4):pdb prot5409. [PubMed] [Google Scholar]
30. Rho M, Tang H, Ye Y. FragGeneScan: predicting genes in short and error-prone reads. Nucleic Acids Res. 2010;38(20):e191. doi: 10.1093/nar/gkq747. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
31. Wommack KE, Bhavsar J, Ravel J. Metagenomics: read length matters. Appl Environ Microbiol. 2008;74(5):1453–1463. doi: 10.1128/AEM.02181-07. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
32. White RA, Blainey PC, Fan HC, Quake SR. Digital PCR provides sensitive and absolute calibration for high throughput sequencing. BMC Genomics. 2009;10:116. doi: 10.1186/1471-2164-10-116. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
33. Adey A, Morrison HG, Asan Xun X, Kitzman JO, Turner EH, Stackhouse B, MacKenzie AP, Caruccio NC, Zhang X, Shendure J. Rapid, low-input, low-bias construction of shotgun fragment libraries by high-density in vitro transposition. Genome Biol. 2010;11(12):R119. doi: 10.1186/gb-2010-11-12-r119. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
34. Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP, Evers DJ, Barnes CL, Bignell HR, Boutell JM, Bryant J, Carter RJ, Keira Cheetham R, Cox AJ, Ellis DJ, Flatbush MR, Gormley NA, Humphray SJ, Irving LJ, Karbelashvili MS, Kirk SM, Li H, Liu X, Maisinger KS, Murray LJ, Obradovic B, Ost T, Parkinson ML, Pratt MR. et al. Accurate whole human genome sequencing using reversible terminator chemistry. Nature. 2008;456(7218):53–59. doi: 10.1038/nature07517. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
35. Nakamura K, Oshima T, Morimoto T, Ikeda S, Yoshikawa H, Shiwa Y, Ishikawa S, Linak MC, Hirai A, Takahashi H, Altaf-Ul-Amin M, Ogasawara N, Kanaya S. Sequence-specific error profile of Illumina sequencers. Nucleic Acids Res. 2011;39(13):e90. doi: 10.1093/nar/gkr344. [PMC free article] [PubMed] [CrossRef] [Google Scholar]
36. Hess M, Sczyrba A, Egan R, Kim TW, Chokhawala H, Schroth G, Luo S, Clark DS, Chen F, Zhang T, Mackie RI, Pennacchio LA, Tringe SG, Visel A, Woyke T, Wang Z, Rubin EM. Metagenomic discovery of biomass-degrading genes and genomes from cow rumen. Science. 2011;331(6016):463–467. doi: 10.1126/science.1200387. [PubMed] [CrossRef] [Google Scholar]