

# Biobank System

## 1 Introduction

From the introduction in the assessment file it could be concluded that a minimum viable product regarding a biobank system should be developed. In terms of the biobank system it is generally an administrative interface for a system allowing a Tissue Sample directory to keep track of the number of samples with certain characteristics contained within a larger collection of samples. In this document I have included specific progress related to this project including requirements gathering, technical considerations, data considerations, implementation, testing, implementation demo, future work, as well as all the documentation needed to execute this project.

## 2 Requirements Gathering


### 2.1 Stakeholder Analysis

Based on the given brief only one stakeholder was identified: a biomedical specialist working with biomedical samples will be considered as primary stakeholder.

### 2.2 Persona

In order to elicit requirements and better understand the stakeholders, I have created one persona for the identified user group:

Table 1 A persona's information regarding user group

	<b>Name</b>	May Stringer
	<b>Gender</b>	Female
	<b>Age</b>	26
<b>Occupation</b>	Biomedical specialist	
<b>Computer Literacy</b>	Comfortable with spreadsheets, Heavy computer user	
<b>Hobbies</b>	Dancing, Jogging	

<b>Bio</b>	University graduate with high literacy skills and many years of experience working with technologies in various biological laboratories.
------------	--

The information regarding the persona was conceived and generated according to the stakeholder analysis in previous step as well as the introduction in the assessment file. Though there is no need to generate a specific user in this product, it is worthy to conceive a persona as it could help tracing the usage of the product or do some other analysis work like analyzing the ease and users' satisfaction of the product. Additionally, generating a persona in advance could also help extend and complete the product in later stages of the development.

### 2.3 Use Case Diagram

In the requirements of the assessment file, there are altogether four tasks to be finished:

- Display a list of collections on the home page (including their title and associated disease).
- Drill into a collection's record to view the details of their currently associated samples.
- Add a new sample to an existing collection.
- Create a new collection.

According to the tasks above and the user analysis in precious stage, the use case was generated as below:

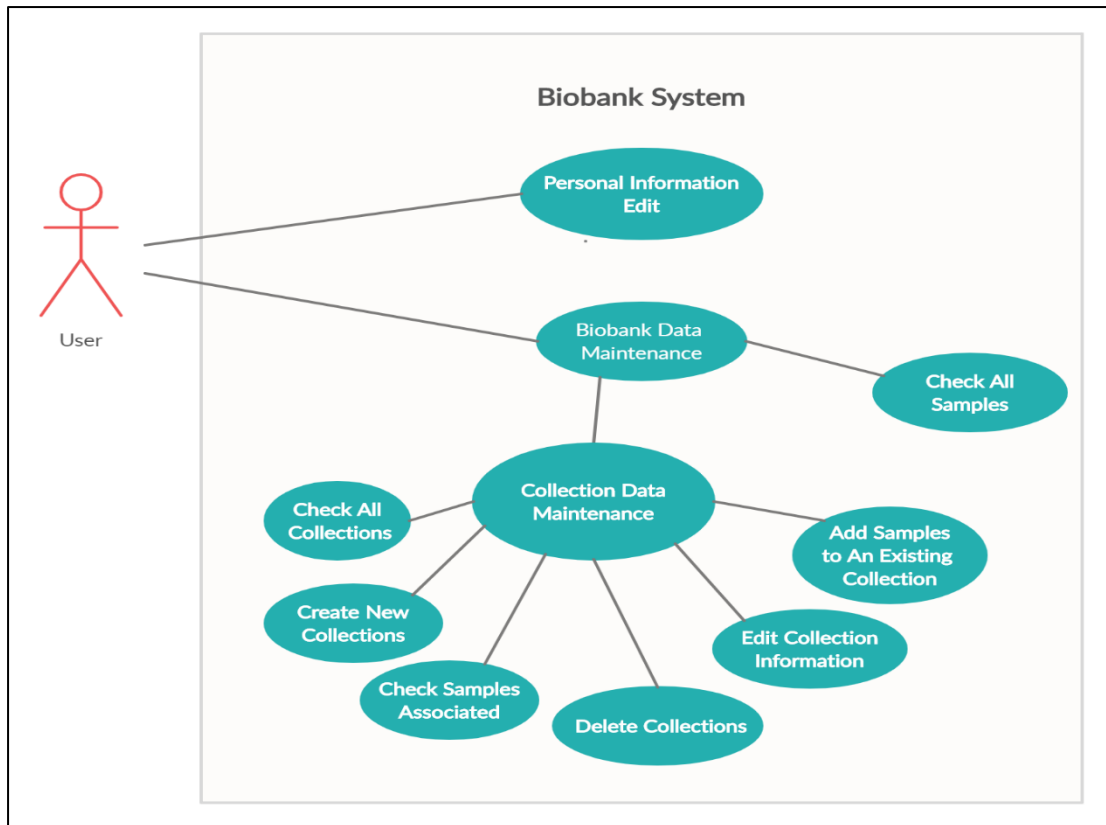


Figure 1 Use Case Diagram

Note this diagram also includes some other use cases which are not part of the brief, including checking all the samples, deleting collections, editing collections' information, and editing personal information.

## 2.4 User Stories

**US1:** As a biomedical specialist working in the lab, I want to be able to view the full list of collections on the home page (including their title and associated disease).

**US2:** As a biomedical specialist I want to review certain samples from different collections, so I want to be able to view all the associated samples to any of the selected collection.

**US3:** As a biomedical specialist I want to be able to add new samples to certain existing collection.

**US4:** As a biomedical specialist I want to be able to add new collections.

## 2.5 List of Requirements

**Requirement 1:** Based on US1 and Persona, it could be concluded that I can implement a table view on the homepage to display all the existing collections.

**Requirement 2:** Based on US2 and Persona, it could be concluded that I can implement a table view on a new page to display all the existing samples regarding each collection.

**Requirement 3:** Based on US3 and Persona, it could be concluded that I can implement a form view on a new page helping users add new samples regarding a specific existing collection.

**Requirement 4:** Based on US4 and Persona, it could be concluded that I can implement a form view on a new page helping users add new collections.

### **3 Technical Considerations**

#### **3.1 Coding Conventions**

To ensure high quality development and long-term maintenance we are using coding conventions. The main programming language adopted in the development of this project was Java, thus this project generally followed the Java language coding standards presented in the Java Language Specification. Notably, the main conventions in this project could be listed below:

1) Generally, all the names including functions' names, variables' names, and classes' names were generated in the form of CamelCase and under the consideration of the ease of understanding.

2) The structure of the files including the packages, java classes, and other files was set for the ease of maintenance and future extensibility.

#### **3.2 Development device and version control**

As stated before, the programming language adopted fulfilling this system was Java 1.8. The development device was Acer - Aspire E5-573G with Windows 10, and the available OS memory was 12208MB RAM.

For better cooperation with other developers in later stages and the long term management of the project source code, the version control tool Github was adopted in this project to record and update the progress of the development as well as backup and share the source code.

#### **3.3 Database considerations**

As there is no enough information provided in the assessment file regarding the prospected size of the data and other requirements like the fund provided by the holders, the required type regarding programming languages, storage engines, platforms, and the needs of backup and security, the database temperately adopted in this product was MySQL 8.0.22 according personal preference and proficiency.

### 3.4 Agile approach

Agile software development refers to software development methodologies centered around the idea of iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams. Scrum and Kanban are two of the most widely used Agile methodologies, and in our development, we will use these. We will run weekly sprints and work closely for continuous feedback with the client.

As from the brief probably we are going to have a small development team that will use weekly, agile approach was chosen to develop this project.

### 3.5 Usage and server specifications

According to the description of requirements in previous stages, the memory needed regarding both the database and the source were prospected as 2GB.

This needs to be established with the clients, and we currently estimate that 1 server (home server) would be enough to be utilized as the host server to host both the website and the database.

### 3.6 Technologies

The technologies adopted building this platform are listed below:

- 1) Spring Frameworks (Spring MVC + Spring JDBC).
- 2) Java Server Pages (JSP).
- 3) JSP Standard Tag Library (JSTL).
- 4) MySQL Connector Java (JDBC Driver) alongside MySQL.
- 5) Maven (Adopted to manage the dependencies).
- 6) JUnit (Adopted for unit testing).

## 4 Data Considerations

**Security:** The data in this product is going to be encrypted in future stages. The one that could get access to this data should be well supervised and managed in later stages (e.g. only logged in users can access this data). This needs to be discussed with the company and user group.

**Storage location:** This location of the data storage in this project will be in line with certain data protection regulations according to stakeholders' needs (e.g. GDPR in the UK/EU, PISS in China).

**Accessibility:** There are different standards for accessibility, and we will try and create a tool that will allow people with various conditions, including visual impairment to be able to use this software tool.

## 5 Implementation

### 5.1 Structure of the project

According to the results of previous stages, here is the UML diagram regarding the project implementing the system(note some other methods or classes were also constructed and showed in this diagram as they could be adopted in the future stages of the development):

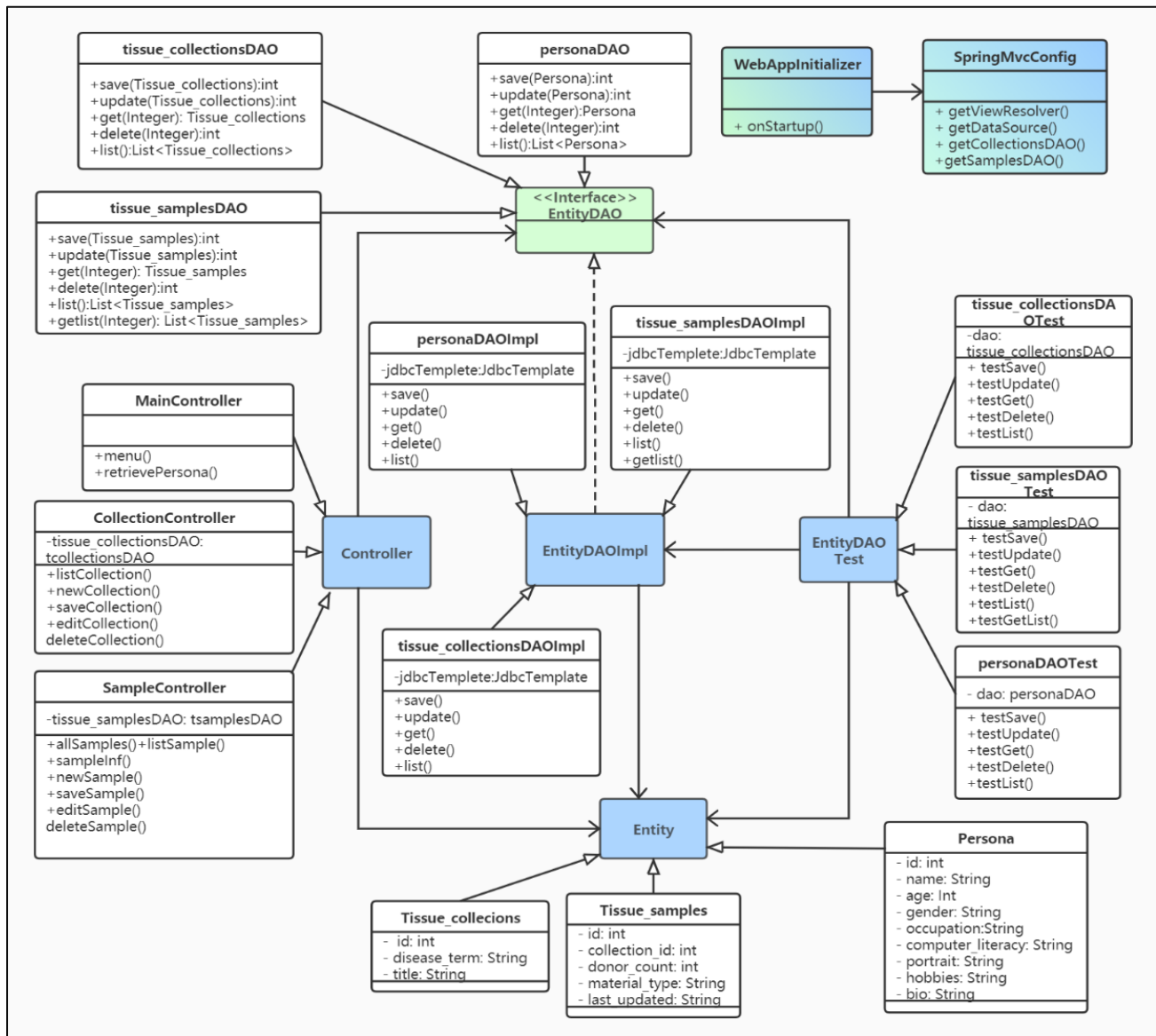


Figure 2 UML Diagram of the project

### 5.2 Software programs Used

According to personal preference and proficiency, the software programs adopted to

development the product are listed below:

- 1) Java Development Kit (JDK): jdk 1.8.0\_251.
- 2) Eclipse IDE for enterprise Java developers: edition - 2020-09(4.17.0).
- 3) Apache Tomcat: version-1.9.
- 4)MySQL Community Server: version-8.0.22.
- 5) Main Dependencies and libraries:
  - a. JavaSE-14.
  - b. javax.servlet-api-4.0.1.jar
  - c. javax.servlet.jsp-api-2.3.3.jar
  - d. jstl-1.2.jar
  - e. mysql-connector-java-5.1.46.jar
  - f. spring-webmvc-4.1.7.RELEASE.jar
  - g. spring-web-4.1.7.RELEASE.jar
  - h. spring-orm-4.1.7.RELEASE.jar
  - i. spring-jdbc-4.1.7.RELEASE.jar
  - j. spring-context-4.1.7.RELEASE.jar

Notably, the reason to choose Maven was it could help easily import all the dependencies or packages needed during the development progress, which could highly save time and improve development proficiency.

Additionally, the Junit was adopted to do the unit test during the development progress guaranteeing the DAO functions and models were constructed correctly and effectively.

### **5.3 Implementation Progress**

**Step1:** Create MySQL database through commend line with MySQL.

**Step2:** Create project in Eclipse IDE.

**Step3:** Specify dependencies (with Maven).

**Step4:** Code Model classes mapping java classes to the table in the database.

**Step5:** Code DAO classes and do unit tests (with JUnit).

**Step6:** Configure Spring MVC Dispatcher Servlet enabling java web application to run on spring MVC.

**Step7:** Configure Spring MVC.

**Step8:** Code Spring Controllers handing the request from the clients.

**Step9:** Code JSPs.

**Step10:** Test the application.

## **6 Test Plan and Results**

In order to make sure that everything was built correctly, I have made a testing plan. I decided to perform some testing on all the implemented requirements in this project:

Table 2 Test Records

Test Id	Test Date & Time	Requirement under Test	Description of what was tested	Input	Expected output	Actual output (Status)
1	2020-10-27 11:50	Requirement 1	Collections view on homepage	Default (no specific input)	A table containing all the collection records stored in the database	Succeeded
2	2020-10-27 12:00	Requirement 1	Collections view on homepage	Default (no specific input)	A table containing all the collection records stored in the database	Succeeded
3	2020-10-27 12:08	Requirement 2	Samples view regarding a specific collection	Click a collection's "sample" action	A table containing all the samples regarding a specific collection is showed	Succeeded
4	2020-10-27 12:12	Requirement 2	Samples view regarding a specific collection	Click a collection's "sample" action	A table containing all the samples regarding a specific collection is showed	Succeeded
5	2020-10-27 12:20	Requirement 3	Add new samples regarding a specific collection	Choose a collection and click "add/edit samples"	A new sample regarding a specific collection is successfully added	Succeeded
6	2020-10-27 12:28	Requirement 3	Add new samples regarding a specific collection	Choose a collection and click "add/edit samples" link	Another new sample regarding a specific collection is successfully added	Succeeded
7	2020-10-27 12:35	Requirement 4	Add new collections	Click "add/edit collection" link	A new collection is successfully added	Succeeded
8	2020-10-27 12:42	Requirement 4	Add new collections	Click "add/edit collection" link	Another new collection is successfully added	Succeeded



## 7. Project Demo

For the sake of the length of this document, only screenshots regarding several significant requirements or functions are showed below:

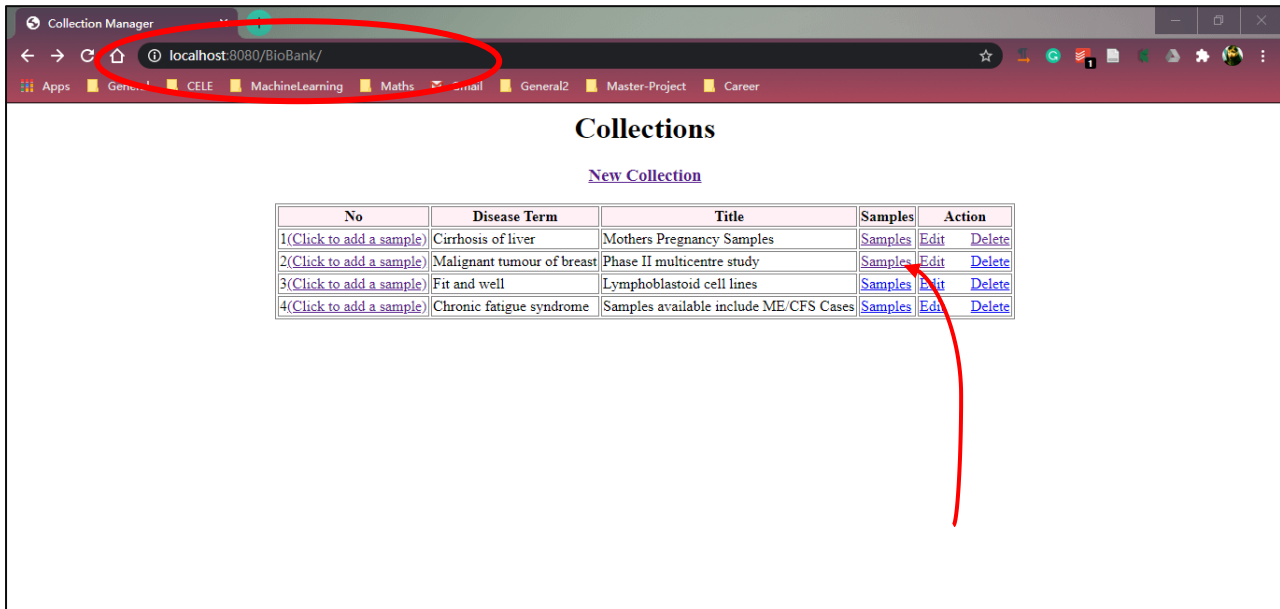
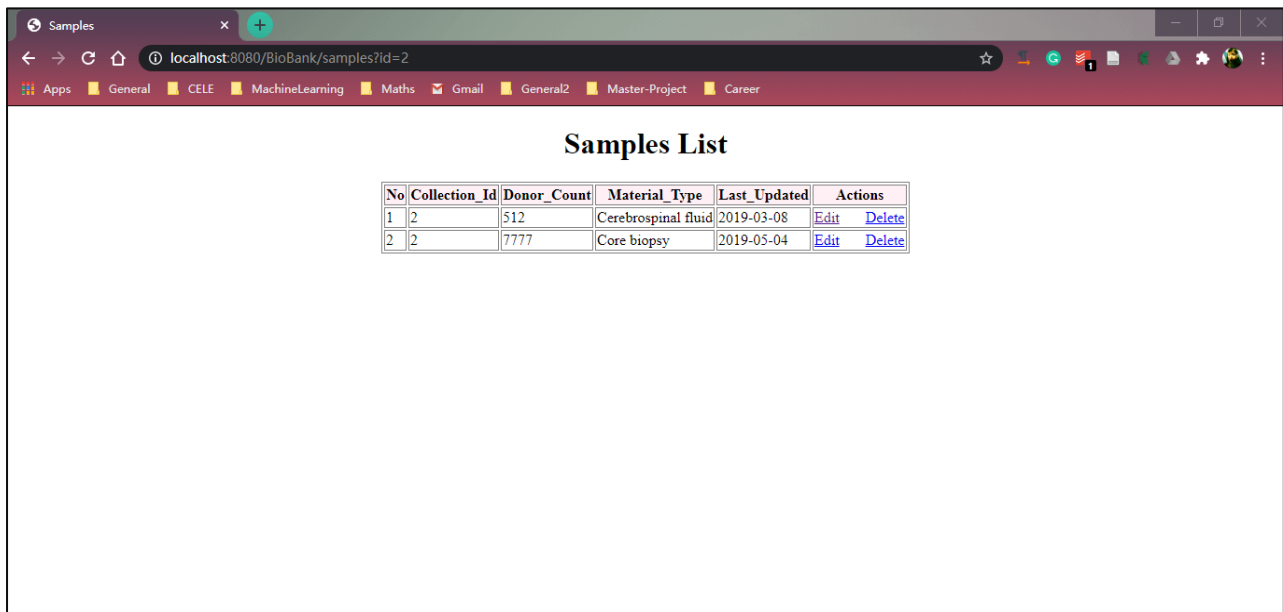


Figure 3. Home Page with Collections list on it

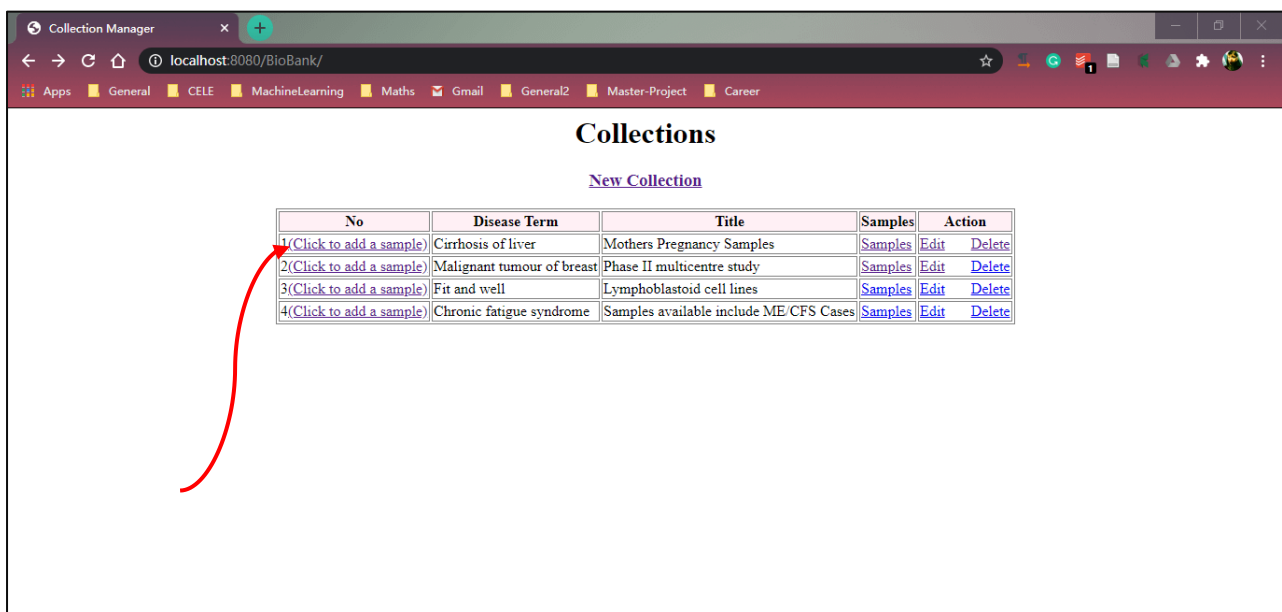
As shown in this figure, when open the home page after inputting the url: <http://localhost:8080/BioBank/>, the list including all the collections will be presented on the home page.

Then click the “Samples” link of each collection, the samples list regarding the chosen collection will be showed. The next figure shows the samples list after choosing the collection with id valued 2.



No	Collection_Id	Donor_Count	Material_Type	Last_Updated	Actions
1	2	512	Cerebrospinal fluid	2019-03-08	<a href="#">Edit</a> <a href="#">Delete</a>
2	2	7777	Core biopsy	2019-05-04	<a href="#">Edit</a> <a href="#">Delete</a>

Figure 4 Show samples regarding collection with id 2



[New Collection](#)

No	Disease Term	Title	Samples	Action
1(Click to add a sample)	Cirrhosis of liver	Mothers Pregnancy Samples	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
2(Click to add a sample)	Malignant tumour of breast	Phase II multicentre study	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
3(Click to add a sample)	Fit and well	Lymphoblastoid cell lines	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
4(Click to add a sample)	Chronic fatigue syndrome	Samples available include ME CFS Cases	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>

Figure 5 Choose a collection to add a sample

On the home page, chose a collection's id and click the link “Click to add a sample”, a new screen where the users could insert a sample to this collection will be shown as below.

**Insert Sample to collection: 1**

Collection id: 1

Donated Count: 9

Material Type: Nylon

Last Updated: 01/10/2020

[Save](#)

Figure 6 Insert a Sample to collection 1

Input the information of a new sample, click “save”. Then a new sample regarding collection 1 will be added to the samples list. The results could be checked below:

**Samples List**

No	Collection_Id	Donor_Count	Material_Type	Last_Updated	Actions
1	1	9	Nylon	2020-10-01	<a href="#">Edit</a> <a href="#">Delete</a>

Figure 7 The results of adding a new sample to collection 1

When coming back to the home page as shown in Figure 3, click the link “New Collection”, a screen where user can insert a new collection will be shown. Note this

screen could also be adopted to revise to current collections' information.

**Insert/Edit a collection**

Disease Term:

Title:

Figure 8 Insert a new collection

Input the information regarding the new collection as above, then click “save”. A new collection will be added to the collections list as shown below:

**Collections**

[New Collection](#)

No	Disease Term	Title	Samples	Action
1(Click to add a sample)	Cirrhosis of liver	Mothers Pregnancy Samples	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
2(Click to add a sample)	Malignant tumour of breast	Phase II multicentre study	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
3(Click to add a sample)	Fit and well	Lymphoblastoid cell lines	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
4(Click to add a sample)	Chronic fatigue syndrome	Samples available include ME CFS Cases	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>
5(Click to add a sample)	Malignant tumour of breast	A randomised placebo-controlled trial	<a href="#">Samples</a>	<a href="#">Edit</a> <a href="#">Delete</a>

Figure 9 The results of adding a new collection

## 8 Future Work

This product is only a small part of the entire system, thus, there are many other functions, requirements, or ways to improve the performance could be considered and implemented in the future:

**Potential Requirement 1:** Adopt 256-bit encryption help protect the data in this system.

**Potential Requirement 2:** For the better maintenance of the database, the trigger should be added to guarantee the stability of the database. For example, when add new samples, the related collection should already exist or the information regarding the related new collection should be added to the collection table.

**Potential Requirement 3:** Construct Log-in portal to only allow logged in users to access this data.

**Potential Requirement 4:** Under the consideration of user-friendliness, a user like a biology specialist should be able to check all the information according to personal preference like sorting all this information based on different factors. Additionally, when adding new collections or samples, there should be drop-down box with existing values as this could help users efficiently input the data.

## 9 The method to execute the project

As limited by the realistic factor that I cannot rent a server right now to deploy the application. The way to run this product should be the same to the way I config and run the application on my personal platform (as the host server).

### 9.1 Download and Install Software

Before executing the application, the related environment should be prepared. First, the platform for executing the application should be windows system with a search engine like Google Chrome. Additionally, the software programs mentioned in Sec 5.2 should be downloaded, installed, and configured on personal platform. The other similar editions regarding the software programs could be adopted alternatively. The specific way regarding downloading and installing the software programs could be easily check in their official websites separately. Here are the example links regarding the way downloading and installing the software platforms:

1) Java and Eclipse:

[https://www.ntu.edu.sg/home/ehchua/programming/howto/eclipsejava\\_howto.html](https://www.ntu.edu.sg/home/ehchua/programming/howto/eclipsejava_howto.html).

2) MySQL: <https://www.mysqltutorial.org/install-mysql/>.

3) Tomcat-Apache: <https://crunchify.com/step-by-step-guide-to-setup-and-install-apache-tomcat-server-in-eclipse-development-environment-ide/>.

### 9.2 Download the Source Code Folder and Database File

Open a search engine, search url: <https://github.com/Jinlei-ZHANG/Biobank>. You can access and download all the source code folder zip file named “BioBank.zip” and database file named “biobank.sql” implemented in this product. Decompress the zip

file to obtain the source code folder. Remember the locations of the source code folder and the database file preparing for next stages.

### 9.3 Load the Project

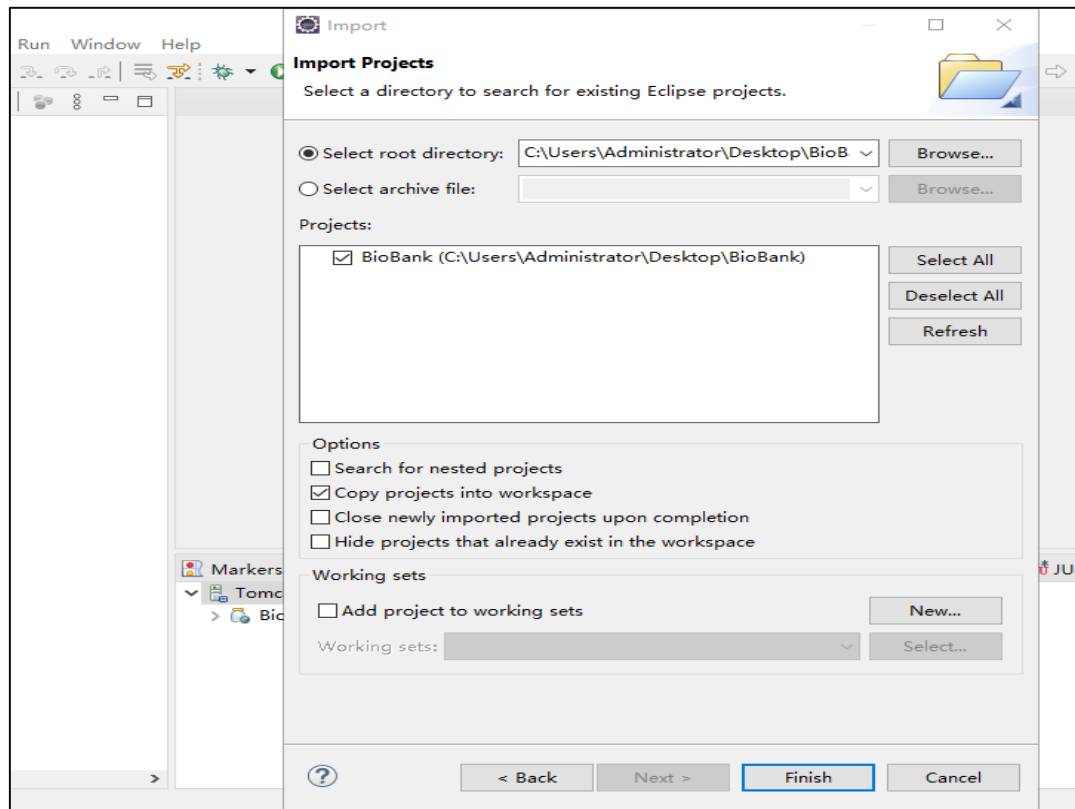


Figure 10 Load source code

Open and get into the main view of Eclipse IDE, click File → Import → General, choose “Existing Projects into Workspace”, you will see a select screen view like above. Click “Browse...” button, find and load the source code folder saved on your computer, then it will automatically return to select screen view, click “Copy projects into workspace” and finish, the source code and related packages will be loaded on your Eclipse IDE.

### 9.4 Load the Database

Open cmd command window and get into the location of your MySQL's bin folder, then input the commands sequentially as below:

- 1) create database biobank;
- 2) use biobank;
- 3) set names utf8;
- 4) source (you path for saving the database file)\biobank.sql;

Figure 12 Load Source Code to Local Host Server

## 9.7 Run the Application (Project)

Finally, open Google Chrome or other search engine, input url: <http://localhost:8080/BioBank/>, you will get into the home page showing like below and could start using the system.

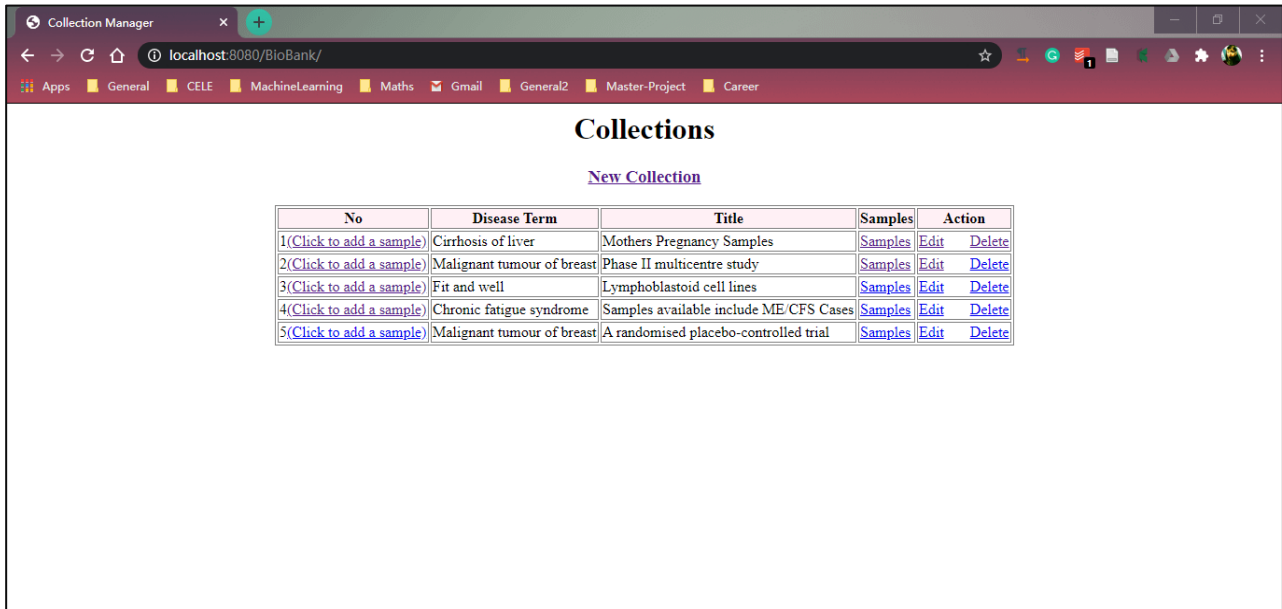


Figure 13 The homepage of the product