

CALIFORNIA STATE UNIVERSITY SAN MARCOS

PROJECT SIGNATURE PAGE

PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE

MASTER OF SCIENCE

IN

COMPUTER SCIENCE

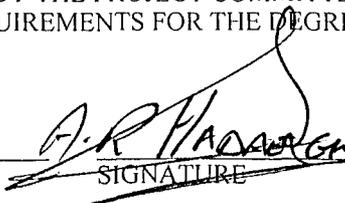
PROJECT TITLE: A REPOSITORY DATABASE SYSTEM TO DO DATA MINING IN DRUG
DISCOVERY

AUTHOR: Jiali Tang

DATE OF SUCCESSFUL DEFENSE: December 2nd, 2019

THE PROJECT HAS BEEN ACCEPTED BY THE PROJECT COMMITTEE IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF
SCIENCE IN COMPUTER SCIENCE.

Dr. Ahmad Reza Hadaegh
PROJECT COMMITTEE CHAIR


SIGNATURE

12/02/2019
DATE

Dr. Xin Ye
PROJECT COMMITTEE MEMBER


SIGNATURE

12/02/2019
DATE

PROJECT COMMITTEE MEMBER

SIGNATURE

DATE

A Repository Database System to do Data Mining in Drug Discovery

Table of Contents

1. INTRODUCTION.....	2
1.1 INTRODUCTION	2
1.2 PURPOSE	2
1.3 DATA MINING PROCESS.....	2
2. SYSTEM OVERVIEW	4
2.1 SYSTEM DESIGN	4
2.3 DATABASE DESIGN.....	5
2.4 WORKFLOW	6
2.5 USER REQUIREMENTS.....	6
3. USER MANUAL.....	8
3.1 HOW TO USE THIS DATABASE SYSTEM	8
3.2 USER INSTRUCTION AND EXAMPLES	8
4. APIS	30
4.1 GETJOB.....	30
4.2 UPDATEJOB	31
4.3 POSTOUTPUT.....	32
5. FUTURE WORK	33
5.1 ENABLE MORE DATA MODELS	33
5.2 DEVELOP THE ADMIN SYSTEM.....	33
5.3 CONNECT WITH DRAGON DIRECTLY	33
5.4 ADD ABSTRACTION LAYER TO SUPPORT DIFFERENT DATA MINING PROGRAMS	33
5.5 IMPROVE THE CLOUD INFRASTRUCTURE.....	33
6. RELATED WORK	34
7. REFERENCE.....	35

1. Introduction

1.1 Introduction

The exponentially increasing amounts of data associated with drug discovery being generated each year make getting useful information from that data more and more critical. With a central repository to keep the massive amounts of data, organizations need tools that can help them extract the most useful information from the data. A data warehouse can bring together data in a single format, supplemented by metadata through the use of a set of input mechanisms known as extraction, transformation, and loading (ETL) tools. Extraction of the data can be either extracting existing data or the data that is imported to the database, transformation is when the data is translated to the format the database can understand. Transformation makes the new format of the data consistent with the other existing data. Finally, the formatted data can be loaded into files and the link address of the data is saved in tables in the database for further analysis. Analysis of the data includes simple query and reporting, statistical analysis, complex multidimensional analysis, and data mining. Large quantities of data are searched and analyzed to discover useful patterns or relationships, which are then used to predict behavior.

1.2 Purpose

The purpose of this project is to produce a repository database of drugs, drug features (properties), and drug targets where data can be mined and analyzed. Drug targets are different proteins that drugs try to bind to stop the activities of the protein. For example, γ -secretase is a protein that causes Alzheimer's. There are certain drugs that can bind to γ -secretase to stop its functionality which in turn may stop Alzheimer's disease. Users can utilize the database to mine useful data to predict the specific chemical properties that will have the relative efficacy of a specific target and the coefficient for each chemical property. This database can be equipped with different data mining approaches/algorithms such as linear, non-linear, and classification types of data modeling. The data models have enhanced with the Genetic Evolution (GE) algorithms [1, 2, through 17]. This paper discusses implementation with the linear data models such as Multiple Linear Regression (MLR) [18], Partial Least Square Regression (PLSR) [19], and Support Vector Machine (SVM) [20].

1.3 Data Mining Process

Data mining is the process of extracting data, analyzing it from many dimensions or perspectives, and then producing a summary of the information in a useful form that identifies relationships within the

data. There are two types of data mining: descriptive, which gives information about existing data; and predictive, which makes forecasts based on the data [23]. This project adopts the predictive approach where data is a matrix of float numbers. The rows refer to the drugs, the columns are the features (the properties) of the drugs, and the criterion of a target is the value of IC50. IC50 represents the compound/substance concentration required for 50% inhibition. Mathematically, we can view this as follows:

$$Y = \beta X \quad \text{which is equal to} \quad Y = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \dots + \beta_n X_n$$

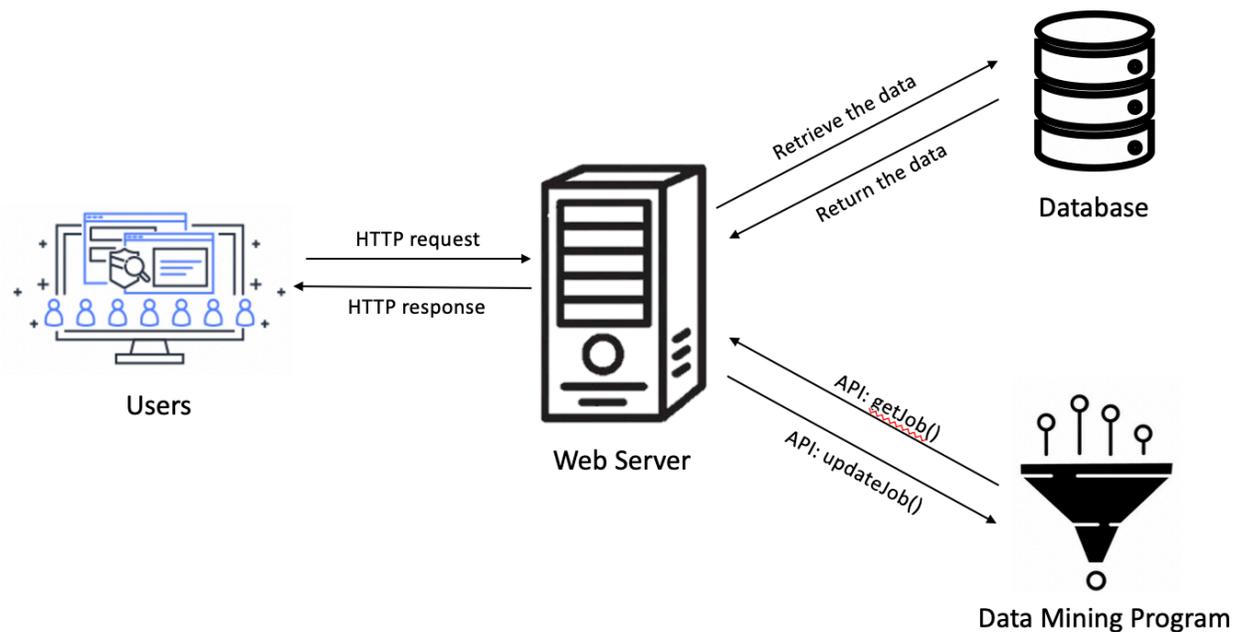
Where Y refers to the value of pIC50 or the actual target. pIC50 is negative log(IC50). Thus, the larger value the pIC50, (or the lower value the IC50 is) the more potent the compound is. In this project, the database is used to train the data using Genetic Evolutionary algorithms: The genetic evolutionary algorithms are Genetic Algorithm (GA)[31], Differential Evolution (DE)[32], Binary Particle Swarm Optimization (BPSO)[12] and hybrid form of DE with BPSO (DE-BPSO)[2 and 7]. This step will select the novel chemical properties by comparing IC50 and ends up with a property set. There are several existing machine learning algorithms that can be used to do the training. However, for this phase of the project, we plan to focus on linear Quantitative Structure Analysis Regression (QSAR)[33] modeling which are PLSR[19], SVM[20], and MLR[18]. This step will identify candidates predicted drug models, which are from the chemical properties and their coefficients resulted from training, to deliver high efficacy in a specific target inhibition. Using Data Mining with Genetic Evolutionary algorithms will eventually improve the quality of the predicted drug models and should ultimately predict the best candidate drug models that can be used by users for drug discovery.

2. System Overview

2.1 System Design

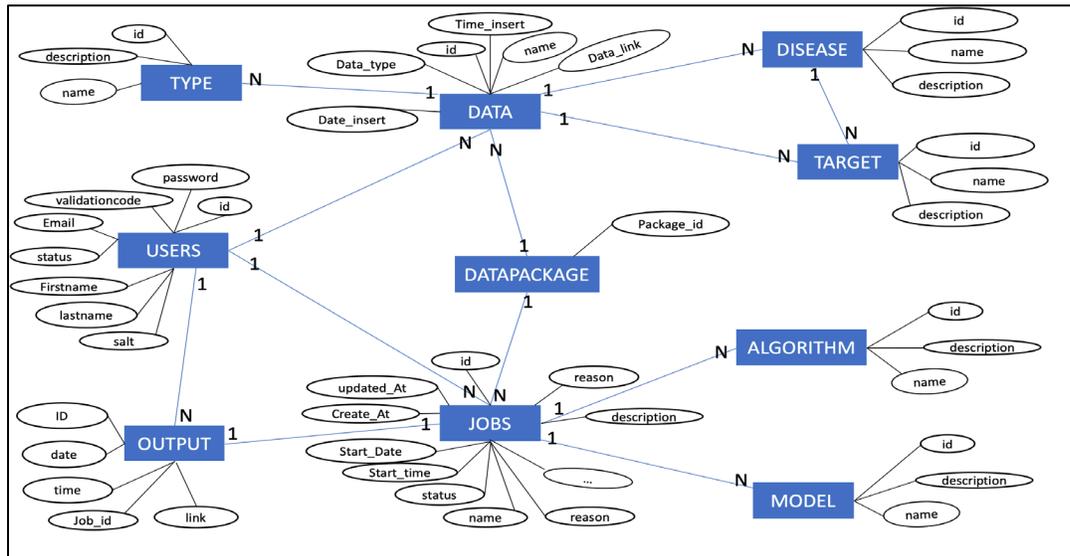
The CSUSMChemoinfo drug discovery database system is developed in PHP and hosted in AWS LightSail Server, which installed Apache web service, MySQL and PHP. The system chooses the relational database MySQL to manage users' information and data. All the data files uploaded by users and the data mining results will be stored in AWS LightSail[34] server for now. This database system will communicate with the other data mining program to execute drug discovery via APIs (Application programming interface). The data-mining program we utilized here is developed by CSUSM Computer Science Department, which is hosted in an AWS EC2 server[35] for now. To support such a machine learning program, we choose a high-performance EC2 instance, which has two CPUs, 4 GiB memory and up to 10 Gigabit network performance.

Flow chart of the elements discussed in the paragraph above. Image displays the user, web server, Database, and Data Mining Program and their interactions.



2.3 Database Design

2.3.1 Database Relationship Design

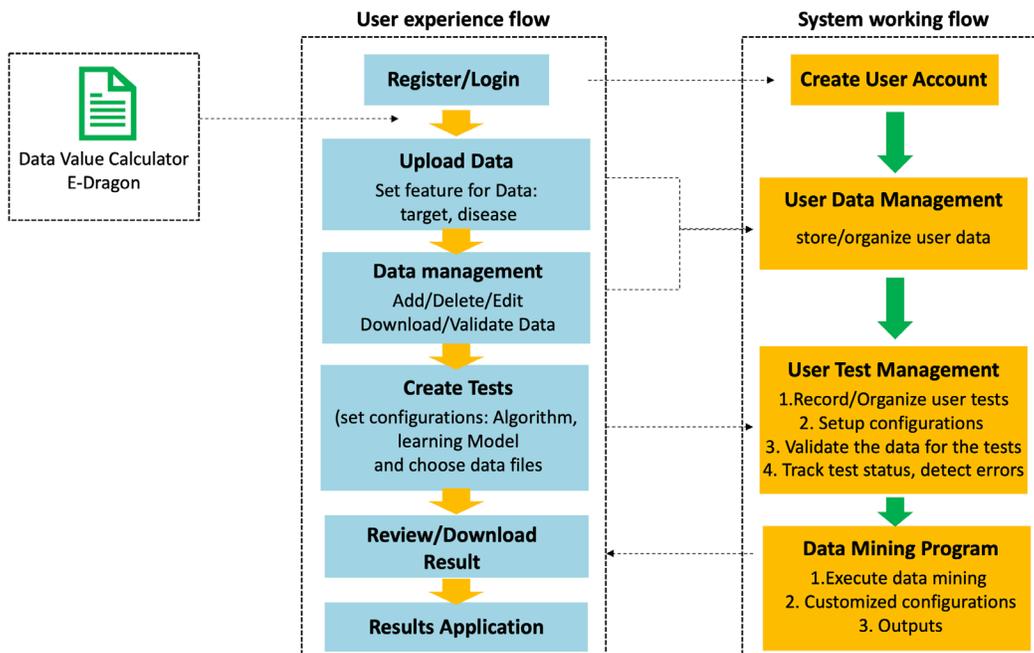


2.3.2 Database Schema

- 1) **USERS:** (user_id, user_password, user_firstname, user_lastname, user_email, user_salt, user_validation_code, user_status)
 user_password is encrypted with user_salt. When a user registers an account, the system will send a link to the user to activate their account. This link is created with the user_validation_code and user_status represents the account is active or not.
- 2) **DATA:** (data_id, data_date_insert, data_time_insert, data_upload_by_user, data_user_id, data_name_by_user, data_link, data_target_id, data_disease_id, data_type)
 The data file uploaded by the user will be saved in the server, and the address of the file will be saved as data_link. Data_type will record what kind of data this file is, there are three types: descriptors, target, and labels.
- 3) **JOB:** (job_id, job_updated_at, job_created_at, job_start_date, job_start_time, job_status, job_user_id, job_name_by_user, job_model_id, job_algorithm_id, job_reason, job_des, data_link, job_attempts, job_queue, job_payload)
 Job_status represents the process of the job, and job_reason is to explain the reason or comment if the job is failed to execute or validate, data_link is to the path of the result.
- 4) **DATAPACKAGE:** (package_id, job_id, data_id)
 This table shows the relationship between data and jobs. A job has multiple data files and a data file can be used for multiple jobs.
- 5) **OUTPUT:** (output_id, output_date, output_time, output_job_id, output_user_id, output_link)
 Output_link will save the address where the results are saved in the server, this record will be updated after a job finishes executing.

- 6) **TYPE:** (type_id, type_name, type_description)
This is the data type. There are three data types: descriptor value, target value, and label.
- 7) **ALGORITHM:** (algorithm_id, algorithm_name, algorithm_description)
An algorithm refers to a data mining algorithm that a user chooses for the execution.
- 8) **MODEL:** (model_id, model_name, model_Description)
A model refers to a data mining model that a user chooses for the execution.
- 9) **DISEASE:** (disease_id, disease_name, disease_description)
An example of the disease is Alzheimer.
- 10) **TARGET:** (target_id, target_name, target_description, target_disease_id)
A target refers to the value of the PCI50s that we explained above.

2.4 Workflow



2.5 User Requirements

Users can use this database system to do drug discovery by uploading their data files and creating data mining tasks. The users can be university professors, students, scientists or researchers.

User can login to the database system to manage their data by uploading, editing, and deleting from the database. Also, they can manage their data mining tasks by creating, editing, deleting from the database, as well as modify their user profile by editing their personal information.

1) Manage account information

- ◇ Register an account
- ◇ Find their password
- ◇ Edit account information (e.g. Name, Email, Password)

2) Manage datasets

- ◇ Upload/Edit/Delete datasets.
- ◇ List all the datasets own by the user.
- ◇ Set configuration about type, disease, target of the data.
- ◇ Check the information about datasets.
- ◇ Search/Download datasets.

3) Manage data mining tasks.

- ◇ Create/Edit/Delete data mining tasks.
- ◇ Set configuration about disease, target, model, algorithm of the task.
- ◇ List all the data mining tasks own by the user.
- ◇ Check the data mining tasks progress and information.
- ◇ Search/Sort tasks by date, name, disease, target, model, algorithm, dataset.
- ◇ Download the results of tasks.

3. User Manual

3.1 How to use this Database System

3.1.1 Register an account/Sign in

In order to use this database system, a user needs to register an account for the first time or sign in after that. To register a new account, a user needs to provide his/her email and sets a password. If the user forgets the password, the system allows the user to receive a new password through his/her email.

3.1.2 Upload Data files

Users are able to upload their data files and manage them in the database system. This system only accepts .csv extension file. Those data files that user uploads will be used to create their data mining test. So, in order to make the data files useful to create a data mining test, it's better that users upload some filtered data, which includes calculated descriptor values, target values, and labels. There are multiple applications that could filter data, such as E-dragon[30]. Filtered data includes three files: calculated descriptor, target, and label. The user can upload these three data files to our database system.

When a user uploads a file, it is required to give it a name, choose what type of disease and target the data file refers to, choose the type of the file and choose the file from their computer.

3.1.3 Create a test

To create a data mining test, there are three data files required: calculated descriptor, target, and label. So before creating a data mining test, a user needs to upload related data files first. After the data files are ready, the user can go to the "My Test" page and click the "Create a new Test" button to create a test. Next, the user is required to give a name, a description for the test, and chooses the algorithm and machine learning model that they expect this data mining test to execute. Then choose three data files for this test from those data files that the user uploaded.

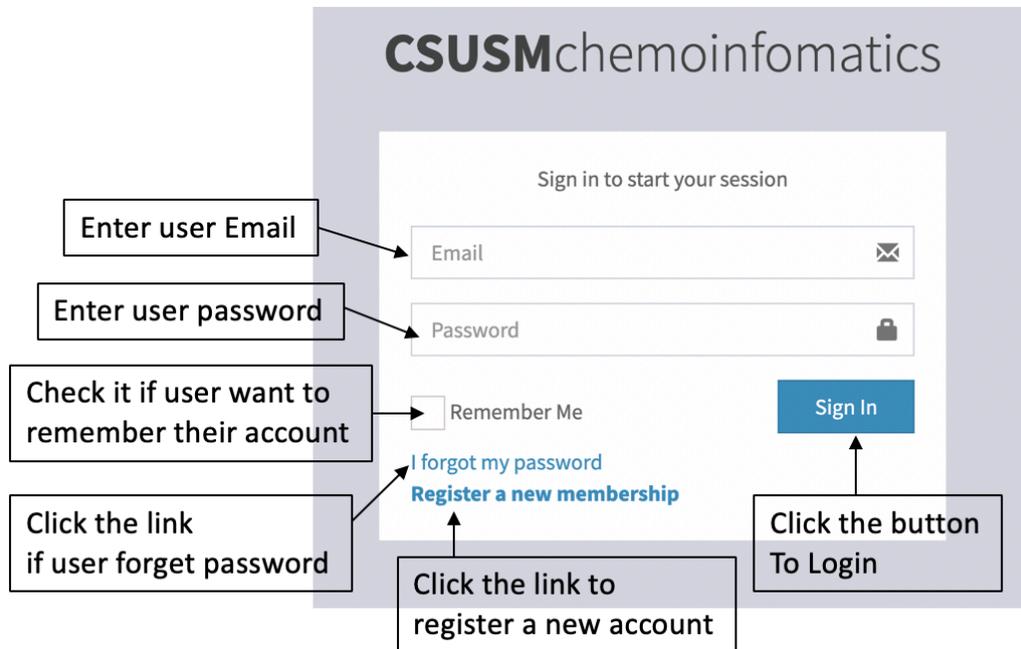
3.2 User Instruction and Examples

3.2.1 Sign in and Register System

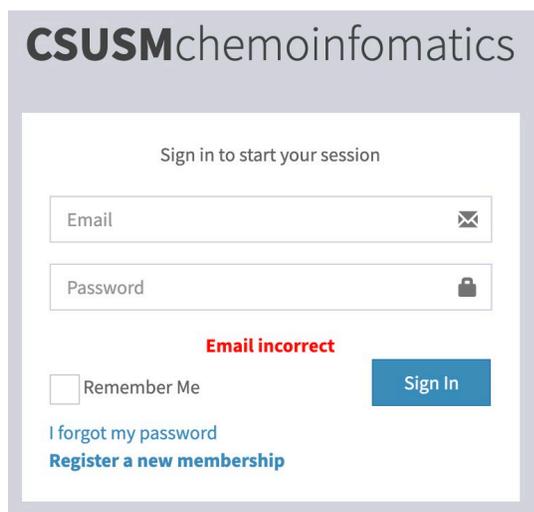
1) *User Sign in*

Sign in page is the first page of the database system application. After a user puts the address in the browser, the system goes to the sign in page first as shown in graph 3.1. If a user has an account already, then he/she can enter the email and password into the correct box, check the "Remember me" button if you want the browser to remember this account, then click the sign in button to the next step. If Email or Password is not correct, it will show an error message as graph 3.2 and 3.3.

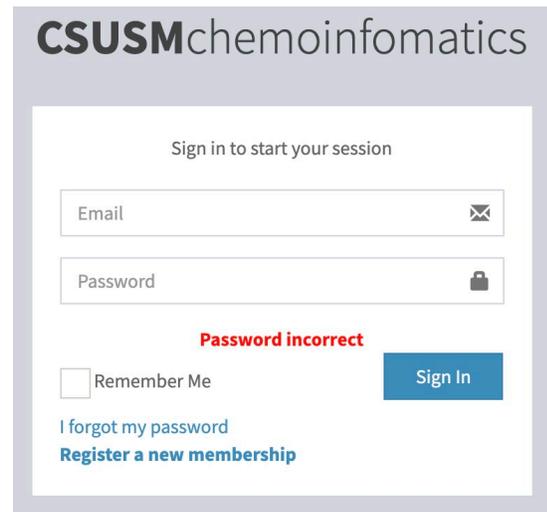
If the user forgets the password, then the user needs to click the link "I forgot my password" and the system will email a link to the user to reset the password. If the user doesn't have an account yet, then the user can click the link "Register a new membership" to register a new account.



3.1 Sign in page



3.2 Sign in email error



3.3 Sign in password error

2) Register a New Account

As graph 3.4 and 3.5 show, the user enters preferred First name, last name, Email, and Password, then re-enters the password to confirm. Agreement on the term is required. Then User can click the “register” button, it will lead to another page tell the user if the application is a success or not (Graph 3.6). The User can follow the instruction to activate their accounts. If the application succeeds, the user

will receive an email from CSUSMChemoinfo with a link in it (Graph 3.7). That's the link for the user to activate their account. The link is unique for every user, and this link will automatically login the user account and leads to the dashboard page. If the Email that the user enters already exist, it will show an error message as graph 3.8.

The image shows the registration form for CSUSMchemoinformatics. The form is titled "Register a new membership" and includes the following fields and elements:

- First name:** A text input field with a person icon on the right.
- Last name:** A text input field with a person icon on the right.
- Email:** A text input field with an envelope icon on the right.
- Password:** A text input field with a lock icon on the right.
- Retype password:** A text input field with a refresh icon on the right.
- Terms:** A checkbox labeled "I agree to the terms" with a link to "terms".
- Register:** A blue button.
- Link:** A link labeled "I already have a membership" with an arrow pointing to the left.

Callouts with arrows point to the following elements:

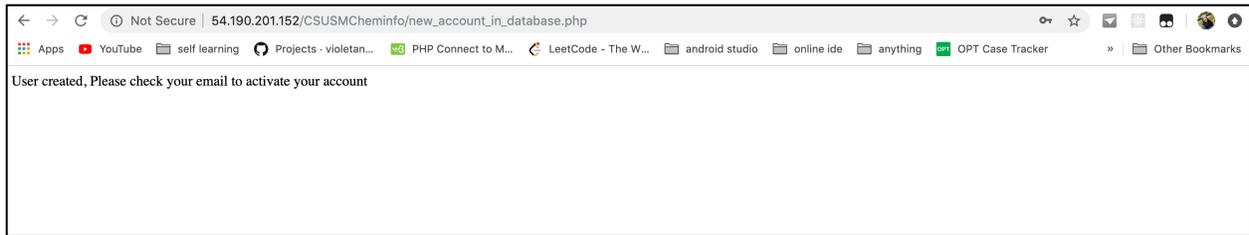
- "Enter user First name" points to the First name field.
- "Enter user Last name" points to the Last name field.
- "Enter user Email" points to the Email field.
- "Enter user password" points to the Password field.
- "Confirm the password" points to the Retype password field.
- "Need to agree with the term Otherwise not able to register" points to the terms checkbox.
- "Click the link go back login page" points to the "I already have a membership" link.
- "Click the button To continue" points to the Register button.

3.4 Register new account page

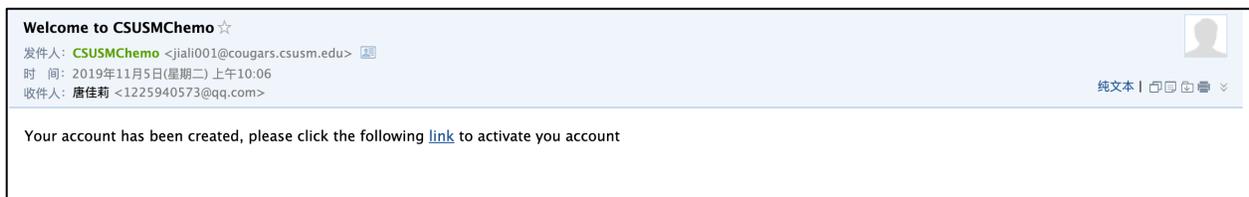
The image shows the registration form for CSUSMchemoinformatics with sample data entered:

- First name:** Joey
- Last name:** TANG
- Email:** thisismyemail@gmail.com
- Password:**
- Retype password:**
- Terms:** I agree to the terms
- Register:** A blue button.
- Link:** A link labeled "I already have a membership".

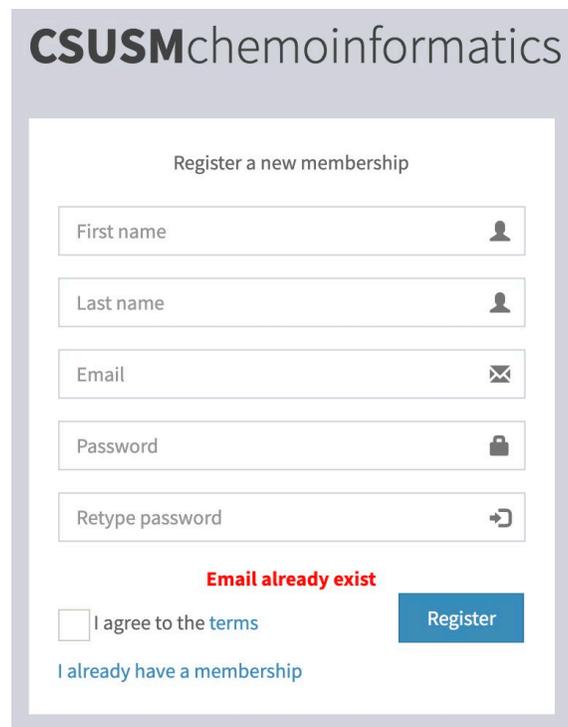
3.5 Register new account example



3.6 Account created notification



3.7 Email send to user to activate their account

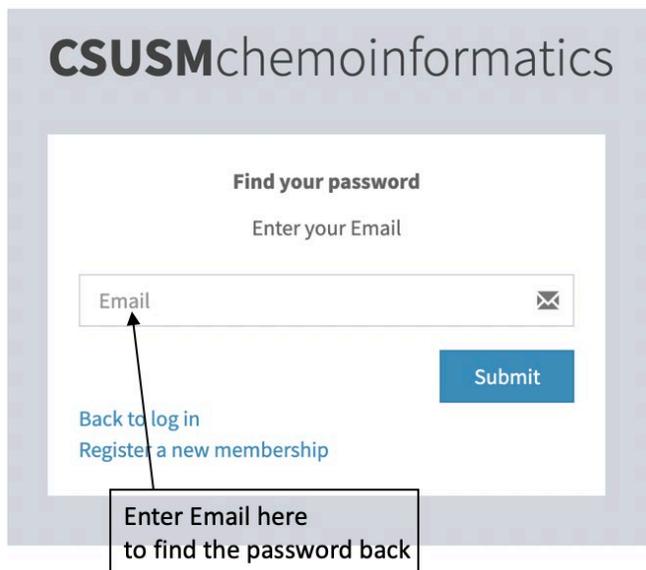


The image shows a registration form for CSUSMchemoinformatics. The form is titled "Register a new membership" and contains the following fields: "First name", "Last name", "Email", "Password", and "Retype password". Below the form, there is a red error message that reads "Email already exist". At the bottom of the form, there is a checkbox labeled "I agree to the terms" and a blue "Register" button. A link "I already have a membership" is also visible at the bottom left of the form area.

3.8 Register error – email already exist

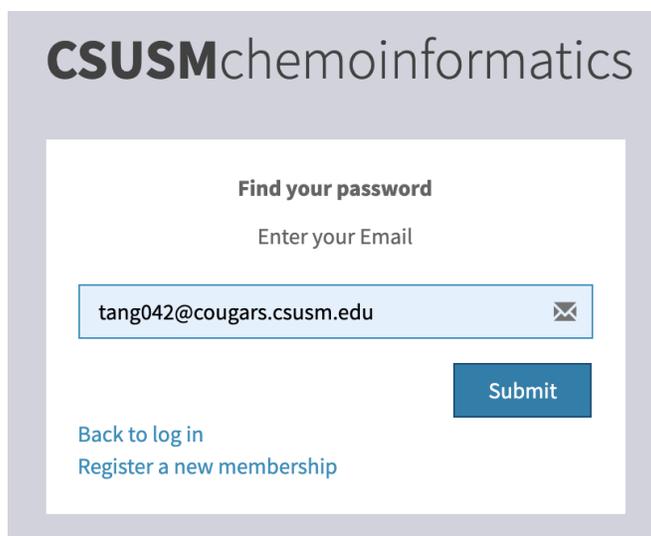
3) Forget/Reset Password

If the user clicks “I forgot my password”, it leads to a page to find their password as it shows in graph 3.9 and 3.10. Then users just need to enter their email and click on the submit button. Next, the system will send an email to the user with a link to reset password and also shows the next step as seen in graph 3.11 which is asking the users to check their email. After a user clicks the link in the email (Graph 3.12), it will lead to a reset password page (Graph 3.13 and graph 3.14).



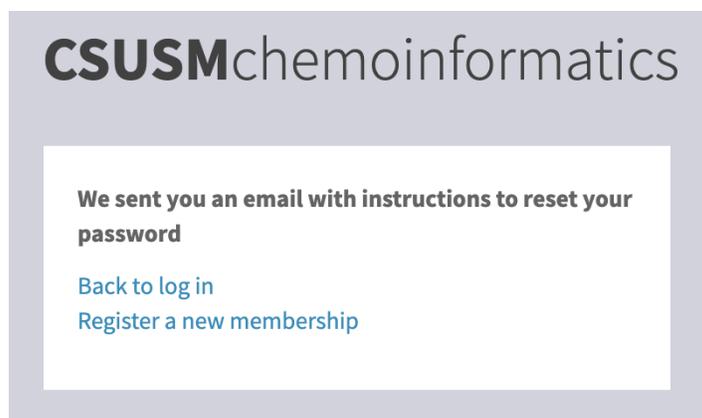
The screenshot shows the 'Find your password' page for CSUSMchemoinformatics. The page has a white background with a grey border. At the top, the text 'CSUSMchemoinformatics' is displayed in a bold, dark font. Below this, the heading 'Find your password' is centered, followed by the instruction 'Enter your Email'. There is a text input field with the placeholder text 'Email' and a small envelope icon on the right. A blue 'Submit' button is positioned to the right of the input field. Below the input field, there are two links: 'Back to log in' and 'Register a new membership'. A callout box with a black border and white background is positioned below the input field, containing the text 'Enter Email here to find the password back'. An arrow points from the top of this callout box to the input field.

3.9 Find password page

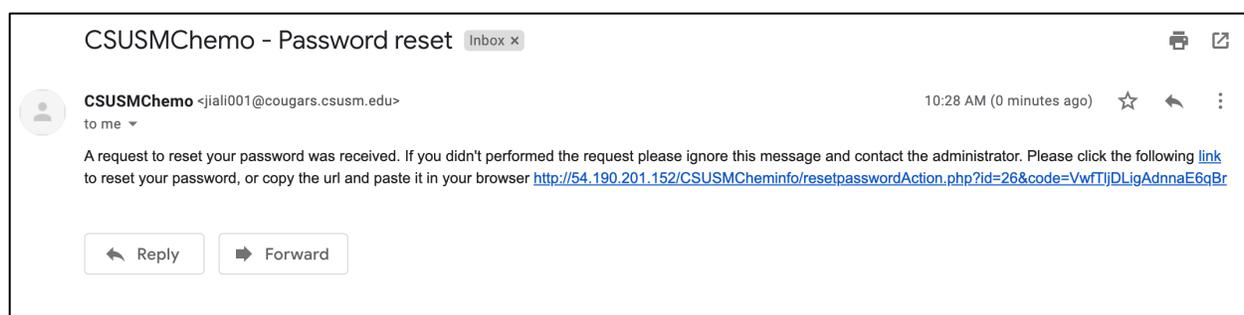


The screenshot shows the 'Find your password' page for CSUSMchemoinformatics, identical to the previous one but with an example email address entered. The text 'CSUSMchemoinformatics' is at the top. Below it is the heading 'Find your password' and the instruction 'Enter your Email'. The text input field now contains the email address 'tang042@cougars.csusm.edu' and has a light blue border. The blue 'Submit' button is to the right. Below the input field are the links 'Back to log in' and 'Register a new membership'.

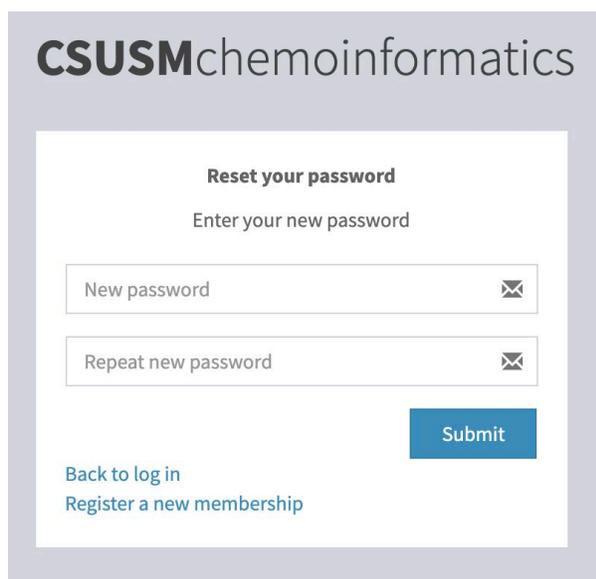
3.10 Find password example



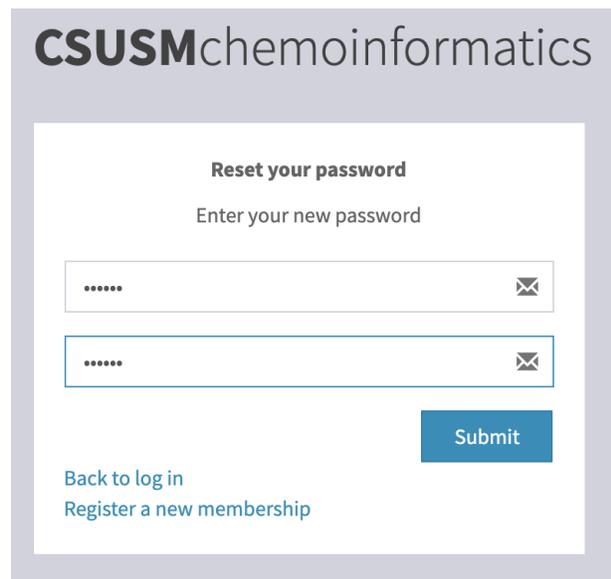
3.11 Notification after enter email



3.12 Email sent to user for reset password



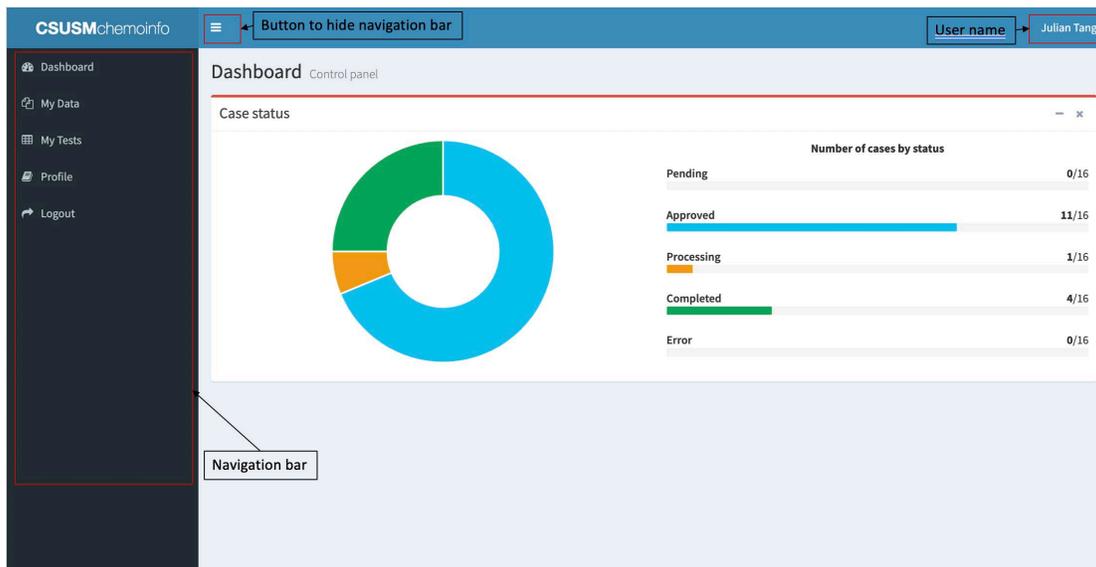
3.13 Reset password



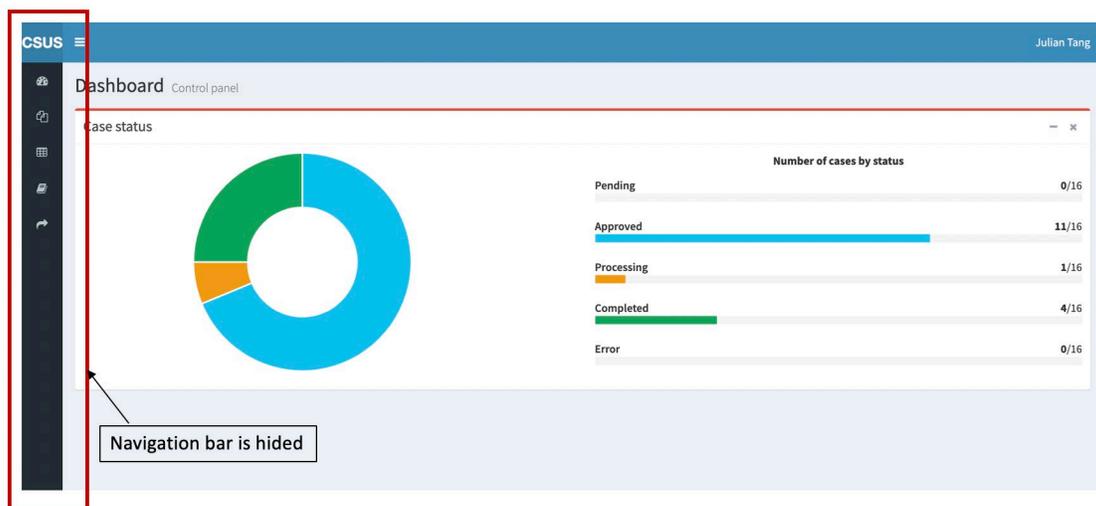
3.14 Reset password example

3.2.2 Dashboard

The dashboard page is a summary page. It summarizes the user's tests status and shows the percentage of different status. So, users can trace their test progress. On the left side, it's the navigation bar. By clicking the navigation icons, we will lead to corresponding pages. "My data" will lead to the user data management page, "My Test" will lead to the test management page, "Profile" will lead to the personal information page and the "Logout" page will logout the current user's account and goes back to the sign in page. On the top, there is a button can help the user to hide the navigation bar (as graph 3.15 and 3.16), and the right corner of the page will show the user name.



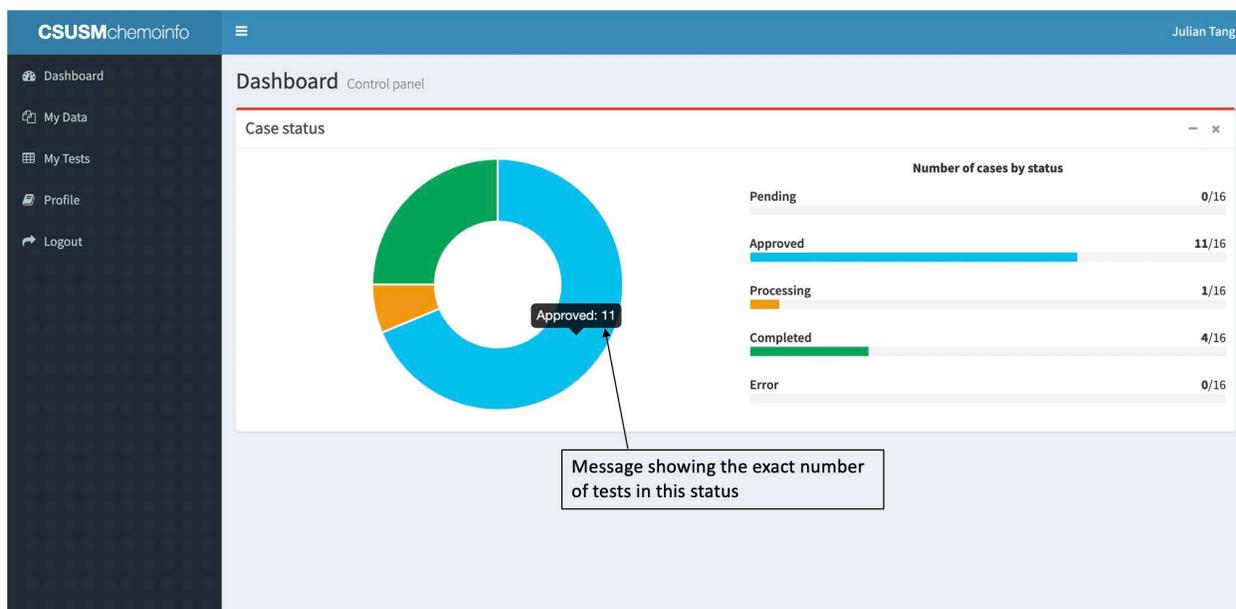
3.15 Dashboard



3.16 Navigation bar is hidden

1) Case status

As shown in graph 3.17, the Case status has two parts, a circle graph, and a progress line table. In the circle graph, the colors represent the status, so it will clearly and visually show the percentage of different statuses, and a user can realize what status of most cases are and if there is an error test. The progress line table will exactly show the number of tests in each status. And if the mouse moves to any color of the circle graph, a message will show up about the exact number of that status.



3.17 Dashboard - Case status

3.2.3 Data management

1) Data List

This Data list page will list all the data file that user uploaded with some basic information, as data name, data type, disease, target and uploaded date. (Graph 3.18) Users can easily download the data file Graph 3.20 or delete the data file by just click the icons Graph 3.21. Also, the data name is clickable graph 3.19, it will lead to a more detailed data information page. If the user wants to upload a data file, then just click the “Upload data” button, which will lead to the upload page for the next instruction.

Users can also find the related data file by keyword search. For example, in graph 3.22, the keyword is “descriptor”, so the list shows all the data files that are Descriptor values type.

CSUSMchemoinfo Jiali Tang

Dashboard | My Data | My Tests | Profile | Logout

Upload Data Dashboard > My Data

My Data

Show 10 entries Search:

#	Data Name	Type	Disease	Target	Upload Date	Download	Delete
1	New Data changed	Lables	HIV	Integres - HIV	11/11/2019		
2	New Data	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
3	New Data	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
4	1	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
5	2	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
6	3	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
7	Alzheimer -des	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/19/2019		
8	Alzheimer -label	Lables	Alzheimer	gamma secretase - Alzheimer	11/19/2019		
9	Alzheimer -target	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/19/2019		

Showing 1 to 9 of 9 entries Previous 1 Next

3.18 Data List fields

CSUSMchemoinfo Julian Tang

Dashboard | My Data | My Tests | Profile | Logout

Upload Data Dashboard > My Data

My Data

Show 10 entries Search:

Data Name are clickable
It will lead to Data detail page

#	Data Name	Type	Disease	Target	Upload Date	Download	Delete
1	New Data	Calculated Descriptor Values	Others	gamma secretase - Alzheimer	11/11/2019		
2	New Data	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
3	New Data	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
4	1	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
5	2	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
6	3	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		

Showing 1 to 6 of 6 entries Previous 1 Next

3.19 Data List - Data names are clickable

CSUSMcheminfo Julian Tang

Dashboard > My Data

Upload Data

My Data

Show 10 entries

Search:

#	Data Name	Type	Disease	Target	Upload Date	Download	Delete
1	New Data	Calculated Descriptor Values	Others	gamma secretase - Alzheimer	11/11/2019		
2	New Data	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
3	New Data	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
4	1	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
5	2	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
6	3	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		

Showing 1 to 6 of 6 entries

Previous 1 Next

54.190.201.152/CSUSMCheminfo/uploads/11573493260.csv

11573493260.csv 1157349207.csv 11573493248.csv

Show All

Download icon are clickable, It will trigger downloading

The file is downloaded immediately

3.20 Data List – download icon

Upload Data

Dashboard > My Data

Can not Delete the Data! This data is used by a test, please check your tests. Data used by a test can not be deleted.

My Data

Show 10 entries

Search:

#	Data Name	Type	Disease	Target	Upload Date	Download	Delete
1	New Data	Calculated Descriptor Values	Others	gamma secretase - Alzheimer	11/11/2019		
2	New Data	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
3	New Data	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
4	1	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
5	2	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
6	3	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		

Showing 1 to 6 of 6 entries

Previous 1 Next

If the Data is being used by a test, then it's not delectable, and an error message will be pop-up

Delete icon is clickable. If the data can be deleted, The data will be deleted directly, otherwise an error message will show up

3.21 Data List – Delete icon and warning notification

Upload Data Dashboard > My Data

My Data

Show 10 entries

Search by keyword: Search: descriptor

#	Data Name	Type	Disease	Target	Upload Date	Download	Delete
1	New Data	Calculated Descriptor Values	Others	gamma secretase - Alzheimer	11/11/2019		
4	1	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		

Showing 1 to 2 of 2 entries (filtered from 6 total entries)

Previous 1 Next

3.22 Data List – search example

2) Show Data Detail / Update data

If the user clicks the data name on the data list page, it will lead to the data detail page, showing more detailed information about the Data file. If this data file is being used in any test, then it cannot be updated or deleted. Showing as graph 3.23, the delete and update buttons are disabled. But if the data file is not being used in any test, then it can be updated or deleted, as graph 3.24.

Data Detail Dashboard > My Data > Data Detail

Data Features of 1

Name: 1

Create Date: 11/11/2019

Create Time: 09:27:28

Data Type: Calculated Descriptor Values

Disease: Alzheimer

Target: gamma secretase - Alzheimer

File Uploaded: [Download](#)

Update Delete

3.23 Data Detail – no editable

Data Features of **New Data**

Name	<input type="text" value="New Data"/>
Create Date	<input type="text" value="11/11/2019"/>
Create Time	<input type="text" value="09:07:57"/>
Data Type	<input type="text" value="Calculated Descriptor Values"/>
Disease	<input type="text" value="Others"/>
Target	<input type="text" value="gamma secretase - Alzheimer"/>
File Uploaded	Download current File <input type="button" value="Choose File"/> No file chosen

3.24 Data Detail – editable

Here is an example, the user is able to change the Name, Data Type, disease, target and update the file. Then the user clicks the update button, then the new information will be saved and go back to the data List Page, as graph 3.25 shows, the Name, Type, disease, Target are changed, and in graph 3.26, the test information is updated to what changed.

Data Detail

Dashboard > My Data > Data Detail

Data Features of **New Data**

Name	<input type="text" value="New Data changed"/>
Create Date	<input type="text" value="11/11/2019"/>
Create Time	<input type="text" value="09:07:57"/>
Data Type	<input type="text" value="Lables"/>
Disease	<input type="text" value="HIV"/>
Target	<input type="text" value="Integres - HIV"/>
File Uploaded	Download current File <input type="button" value="Choose File"/> No file chosen

Annotations: "Those field are editable" points to Name, Data Type, Disease, and Target. "Upload new file" points to the Choose File button.

3.25 Data Detail – update example

My Data

Show 10 entries Information are updated Search:

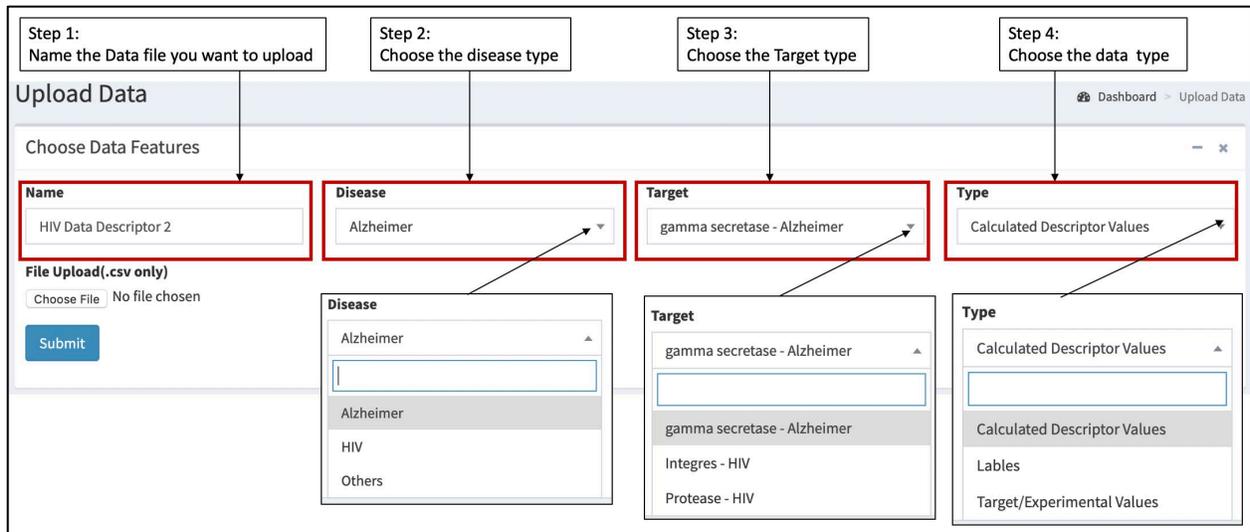
#	Data Name	Type	Disease	Target	Upload Date	Download	Delete
1	New Data changed	Lables	HIV	Integres - HIV	11/11/2019		
2	New Data	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
3	New Data	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
4	1	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
5	2	Lables	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
6	3	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/11/2019		
7	Alzheimer -des	Calculated Descriptor Values	Alzheimer	gamma secretase - Alzheimer	11/19/2019		
8	Alzheimer -label	Lables	Alzheimer	gamma secretase - Alzheimer	11/19/2019		
9	Alzheimer -target	Target/Experimental Values	Alzheimer	gamma secretase - Alzheimer	11/19/2019		

Showing 1 to 9 of 9 entries Previous **1** Next

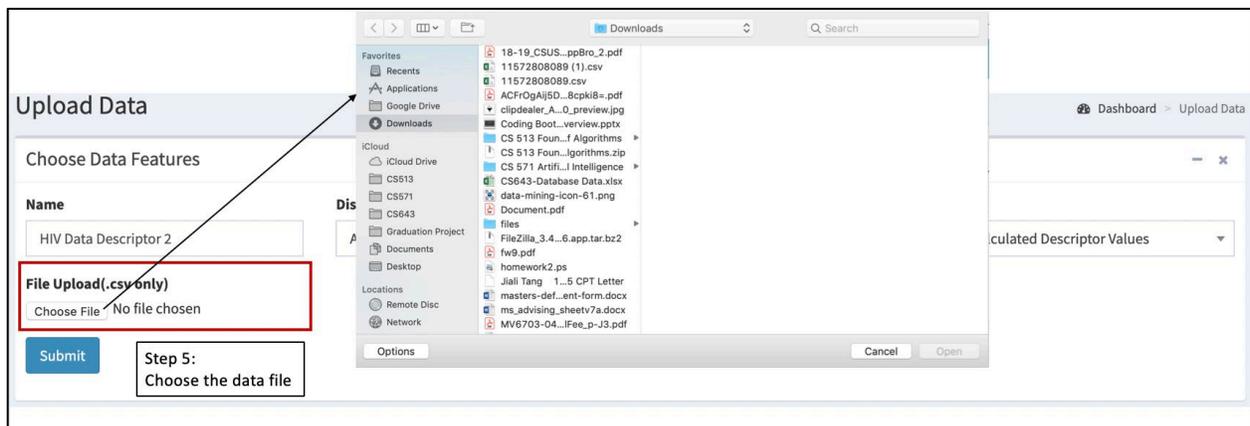
3.26 Data List is updated after Data information is changed

3) Upload Data

In the upload data page, users can set up the configuration of the data file that they uploaded. As graph 3.27 shows, the user can name the data file, and choose what's the disease and target this data used for, also choose the type of this data file. The name field is a text box which is editable, disease, target, and data type fields are a dropdown menu that user can choose from the choices, as it shows in graph 3.27. Right now, this system supports Alzheimer's and HIV diseases. We have done some work for HIV-Protease, HIV-Integrase and gamma-secretases for Alzheimer's. There are three different data types: Calculated descriptor values, target/experimental values, and Labels. Users usually can get those three data files from E-dragon. After setting up those configurations, then click the button "Choose File" as it shows in graph 3.28. The browser will open a window to let the user choose the file that they want to upload. If the user chooses a file, on the right side of the "Choose File" button will show the name of the file that the user chooses. Only .csv file is acceptable. Then click the "Submit" button, the file will be uploaded, and the user can access the file all the time. If the upload fails, an error message will let the user know what the problem is (Graph 3.29).



3.27 Upload Data



3.28 Upload Data – choose file



3.29 Upload Data – Error message

3.2.4 Test management

1) Test List

This Test list page will list all the tests that users created with some basic information, such as test name, test description, algorithm, model, create date and finish date, graph 3.30. Users can easily download the data mining result by just click the download icons. Also, the test name is clickable, it will lead to a more detailed test information page. If the user wants to create a new test, then the user can click the “Create new test button”, which will lead to the create test page for the next instruction.

Test Name are clickable
It will lead to Data detail page

If the test is completed, then the download icon will show here. User can click it to download the result

#	Test Name	Description	Algorithm	Model	Create Date	Finish Date	Status	Result
11	again trip		GA	SVM	11/16/2019		Error	
12	HEACHE		GA	SVM	11/16/2019	11/16/2019	Completed	
13	Alzheimer - created by Jay		GA	SVM	11/16/2019	11/16/2019	Completed	
14	hey try try weird		GA	SVM	11/16/2019		Error	
15	Alzheimer - created for Brain		GA	SVM	11/19/2019	11/19/2019	Completed	
16	test on tuesday		GA	SVM	11/19/2019	11/19/2019	Completed	
17	test on supervisor		GA	SVM	11/19/2019	11/19/2019	Completed	
18	test manually		GA	SVM	11/19/2019	11/19/2019	Completed	
19	test for .sh		GA	SVM	11/19/2019	11/19/2019	Completed	
20	test 20191119		GA	SVM	11/19/2019		Processing	

Showing 11 to 20 of 28 entries

Previous 1 2 3 Next

3.30 Test List

Users can also find the related test by keyword search. For example, in graph 3.31, the keyword is “completed”, so the list shows all the tests that are completed.

The screenshot shows the 'My Test' page in the CSUSMchemoinfo application. A search box is highlighted with a red box and labeled 'Search box'. The search box contains the text 'completed'. Below the search box is a table with 4 entries. The table has columns for #, Test Name, Description, Algorithm, Model, Create Date, Finish Date, Status, and Result. The Status column shows 'Completed' for all entries. The Result column shows a download icon for all entries.

#	Test Name	Description	Algorithm	Model	Create Date	Finish Date	Status	Result
1	test1_tang	hey	GA	MLR_Linear	06/29/2018	10/28/2019	Completed	Download
2	sadjhgipsda	gmjoasjgdsa	GA	MLR_Linear	10/30/2019	11/04/2019	Completed	Download
3	JTest	test	GA	MLR_Linear	10/29/2019	11/04/2019	Completed	Download
4	to test download		GA	MLR_Linear	11/01/2019	11/04/2019	Completed	Download

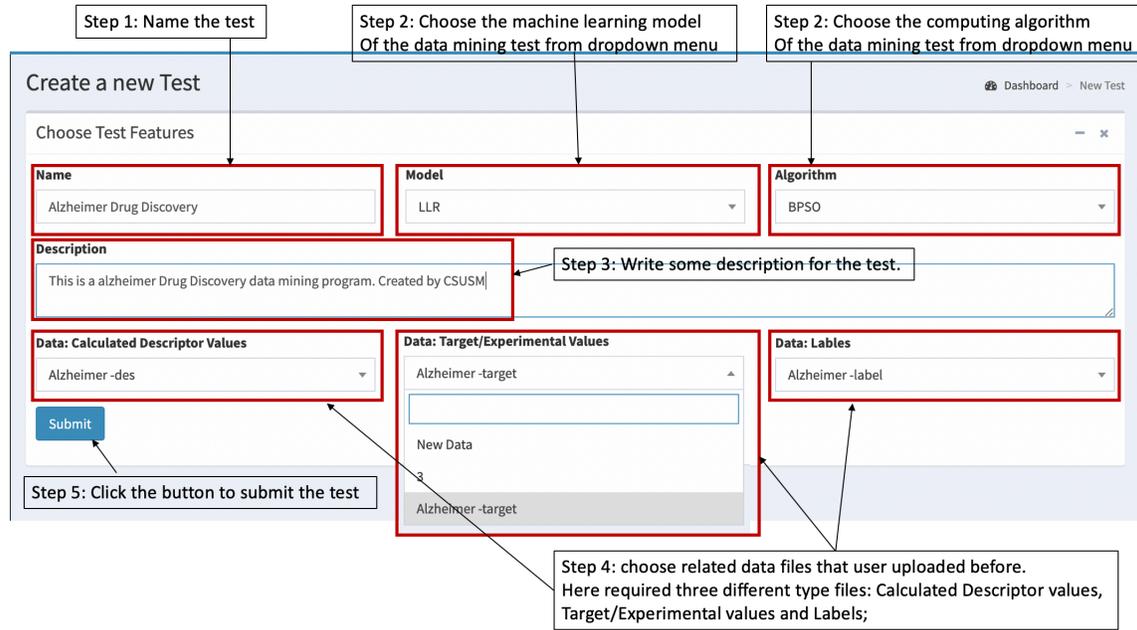
Showing 1 to 4 of 4 entries (filtered from 16 total entries)

Previous 1 Next

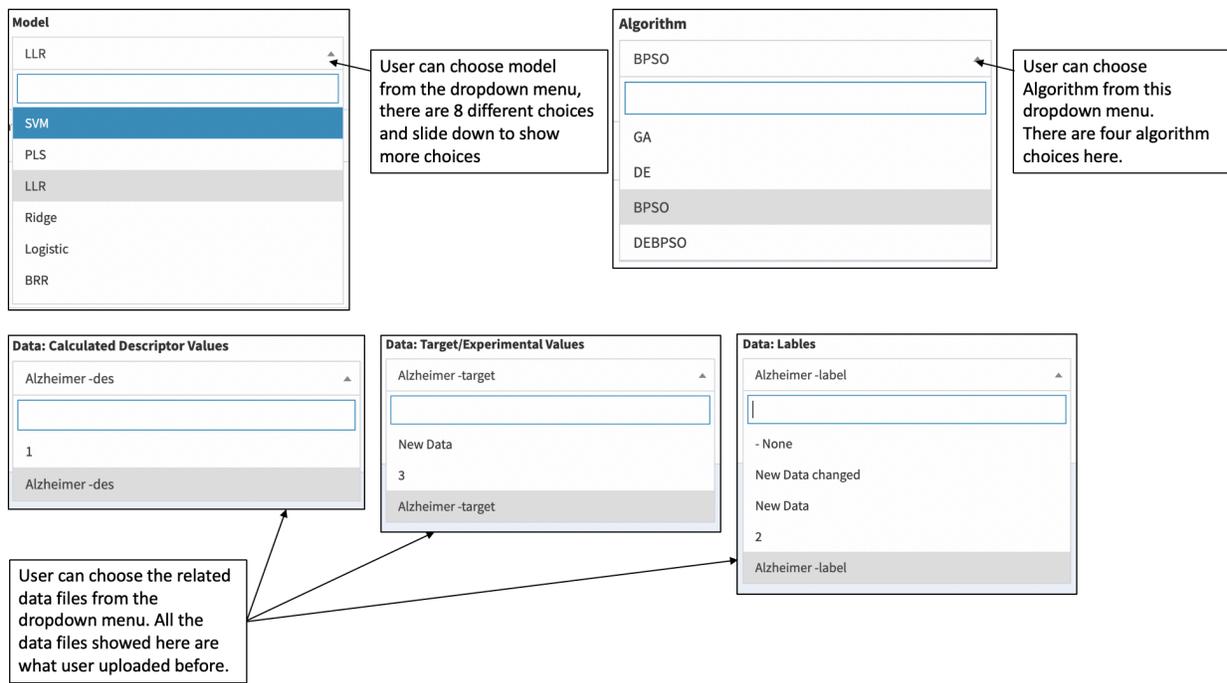
3.31 Test List – search

2) Create Test

In the create test page, users can set up the configuration of the test that they want to create. As graph 3.32, shows, the user can name and describe the test, and choose what's the algorithm and model that they want this data mining test to be computed. Also, the user will need to choose three data files from what they uploaded before. Name and description fields are text box which is editable, algorithm and fields are a dropdown menu that user can choose from the choices (Graph 3.32). There are three different data type files that the user needs to choose: Calculated descriptor values, target/experimental values, and Labels. All the choices listed there are the user's own data files. After setting up those configurations, then the user can click the button "Submit" for the test to be started. When the test is complete, the "Status" will change to "Complete".



graph 3.32 Create a new Test



graph 3.33 Dropdown menus of create a test

3) Test Detail

If users click the test name on the test list page, it will lead to the test detail page, showing more detailed information about the data mining test. If this test is being processed, then it cannot be updated or deleted. As shown in graph 3.34, the delete button is disabled. But if the test is completed, then it can be deleted (Graph 3.35). Further, if the “Status” becomes “Error”, the test can be updated or delete.

In the Test detail page, besides that basic information of the test, data files that used for the test will also be shown. If a user wants to check more information about the data file, the user can click the “More detail” button which will lead to the data detail page. Also, users can click “download” to download the data file directly.

Comment/Error explanation field will explain the reason if a test fails or provide some comment about the failed test.

If the test is completed, then in the download result field, there should be a link that allows the user to download the result file.

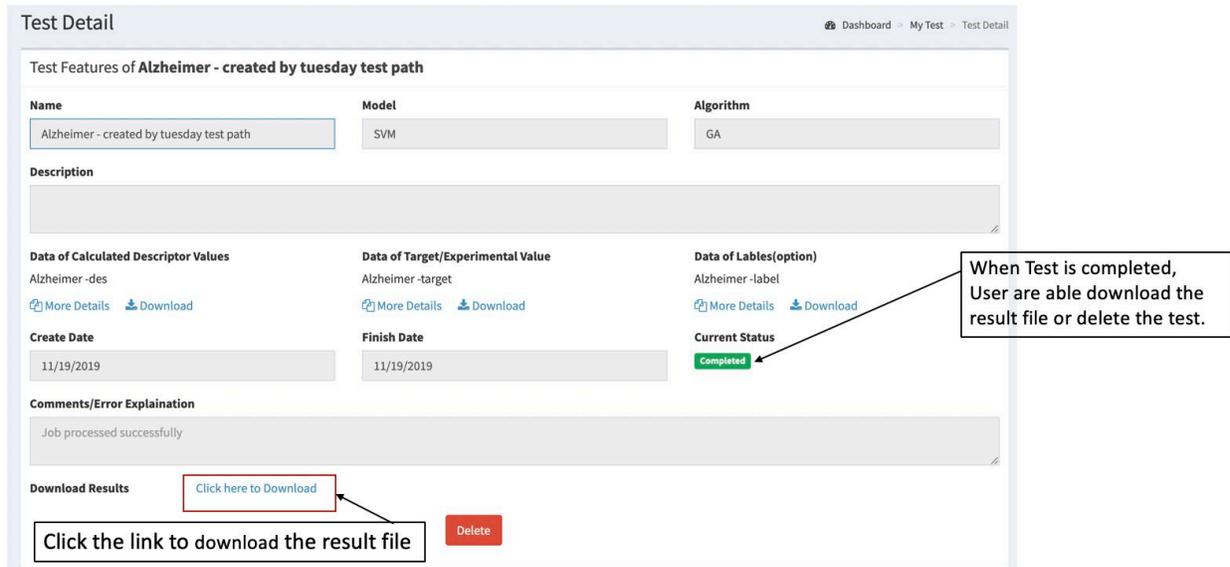
The screenshot shows the 'Test Detail' page for test ID 'test 20191119'. The page layout includes:

- Test Features of test 20191119:**
 - Name:** test 20191119
 - Model:** SVM
 - Algorithm:** GA
- Description:** A large text area for the test description.
- Data of Calculated Descriptor Values:** A table with 1 row and 2 columns. The first column has a 'More Details' link and a 'Download' button.
- Data of Target/Experimental Value:** A table with 1 row and 2 columns. The first column has a 'More Details' link and a 'Download' button.
- Data of Lables(option):** A table with 2 rows and 2 columns. The first row has a 'More Details' link and a 'Download' button. The second row has a 'Current Status' field with the value 'Processing'.
- Create Date:** 11/19/2019
- Finish Date:** none
- Comments/Error Explanation:** A large text area for comments or error messages.
- Download Results:** A section with the text 'Nothing to download yet' and a disabled 'Delete' button.

Annotations on the screenshot:

- A callout box on the right states: "When Test is in process, User are only able to review the information of test, Not able to edit or delete it. So all the form and button here are not editable or disable."
- A callout box at the bottom center states: "Click the link, can go to data file page, check more details of the data. Or click the download button to download the data file directly."

3.34 Test Detail – not editable



3.35 Test Detail – delete

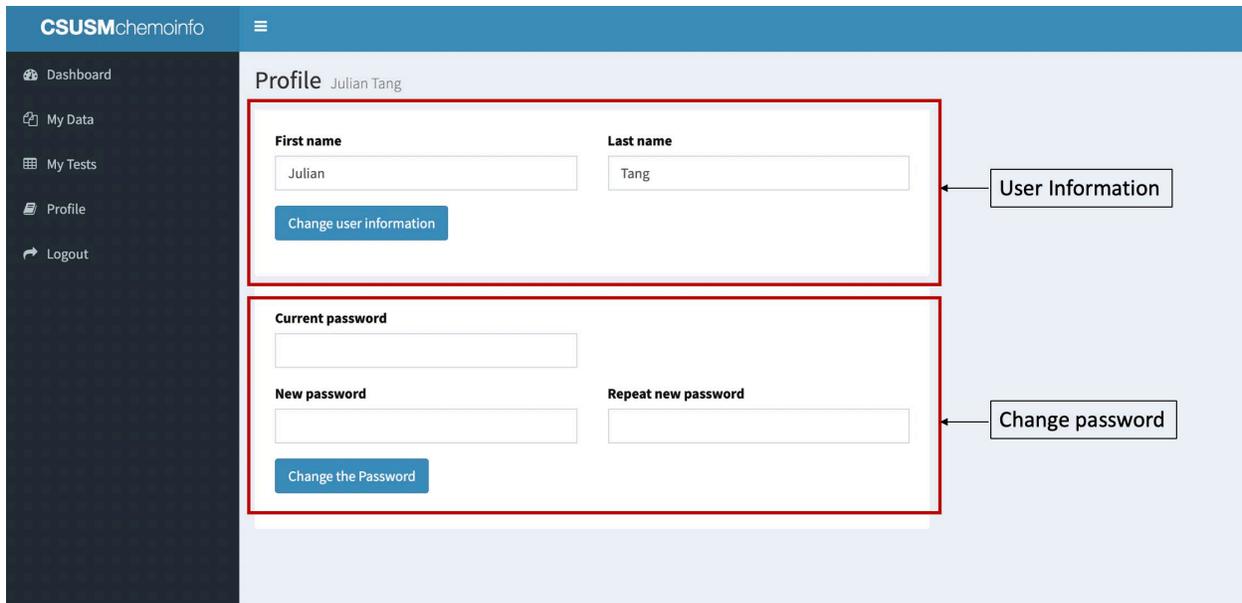
4) Update Test

Sometime, the data mining test may fail because of the incorrect data file or networking problem. So, users can choose to delete the test or reset the configuration of the test and let it be executed again. So, if the test's status is a error, the user can go to the test detail page and check what's the reason for the failure. Based on the error, the user can fix the configuration, such as change the algorithm, change the model, or update correct data files. After editing the test configuration, and clicking the update button, the data-mining test will be sent to the data mining program to execute again.

3.2.5 Profile management

1) Personal Information

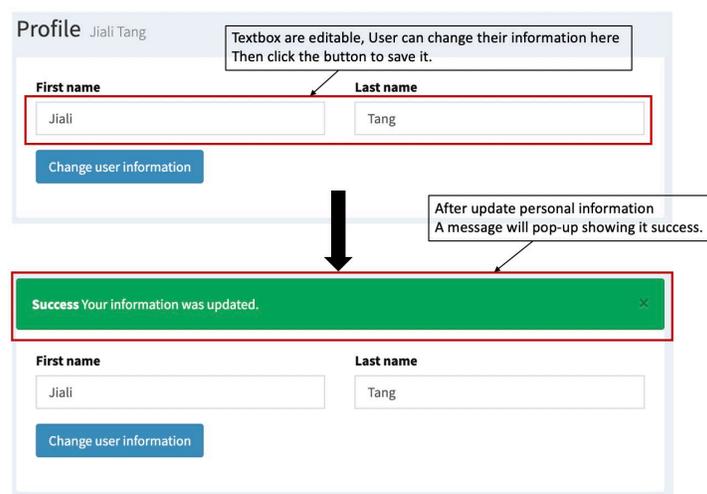
By clicking the "Profile" item in the navigation bar, we will lead to the Profile management page. Here users can see their personal information, such as First name and Last name. Also, Users are able to change the password there. (Graph 3.36)



3.36 Profile page

2) *Change User Information*

In the user information area, both first name and last name are editable. Users can edit them directly and click the button to save it. Then, the success notification will pop up. For example, if a user changes her name from “Julian” to “Jiali”, then click the button “change user information” As we see in Graph 3.37, the name has been updated already and a green notification notifies the user that the information was updated.



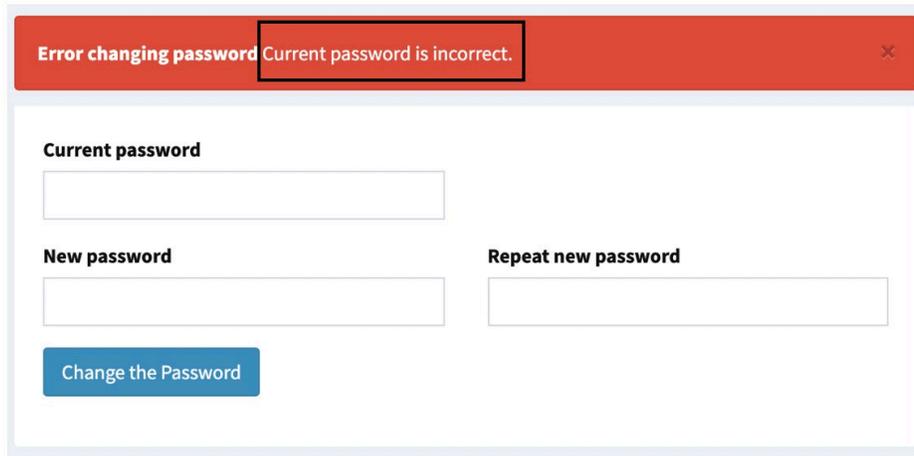
3.37 Change User information

3) Change Password

If a user wants to change the password, the user checks the password area, enters the current password, and the new password. To confirm the new password, the user needs to repeat the new password. Then click the button to update the new password. Here is an example showing how to change the user password: After entering the current password and the new password, then click the button “Change the Password”. As you can see in the graph 3.38, a green notification telling the user that the password is updated. This means the new password is saved. However, if the current password is not correct or the two new passwords do not match, an error notification will pop-up and let the user know what the problem is (Graphs 3.39 and 3.40).

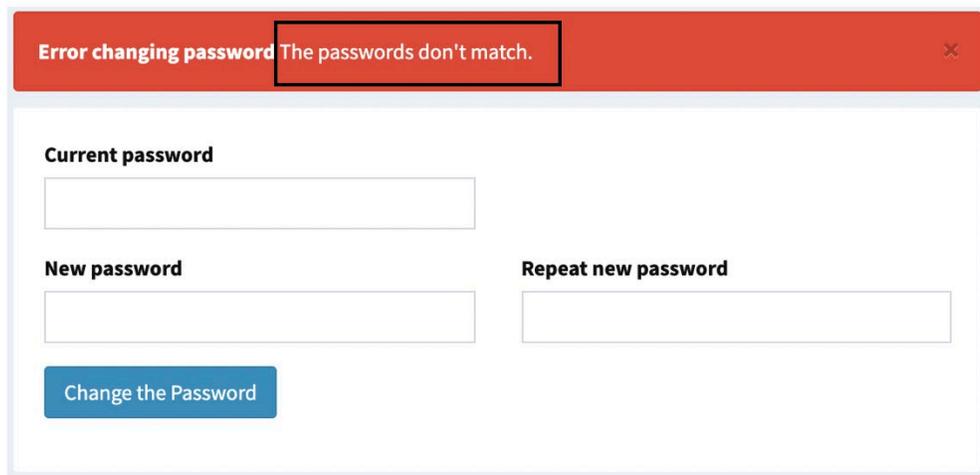
The diagram illustrates the password change process in two stages. The top stage shows a form with three input fields: "Current password", "New password", and "Repeat new password", and a "Change the Password" button. A callout box points to the input fields with the text "Enter the current password and the new password". A thick black arrow points down to the second stage. The second stage shows the same form, but with a green notification bar at the top that says "Success Your password was updated." with a close button. A callout box points to the notification bar with the text "A notification will pop-up and Tell user password is updated."

3.38 Change password



The screenshot shows a web form for changing a password. At the top, a red error banner displays the text "Error changing password Current password is incorrect." with a close button (X) on the right. Below the banner, the form contains three input fields: "Current password", "New password", and "Repeat new password". A blue button labeled "Change the Password" is positioned below the "New password" field.

3.39 change password error- current password is incorrect

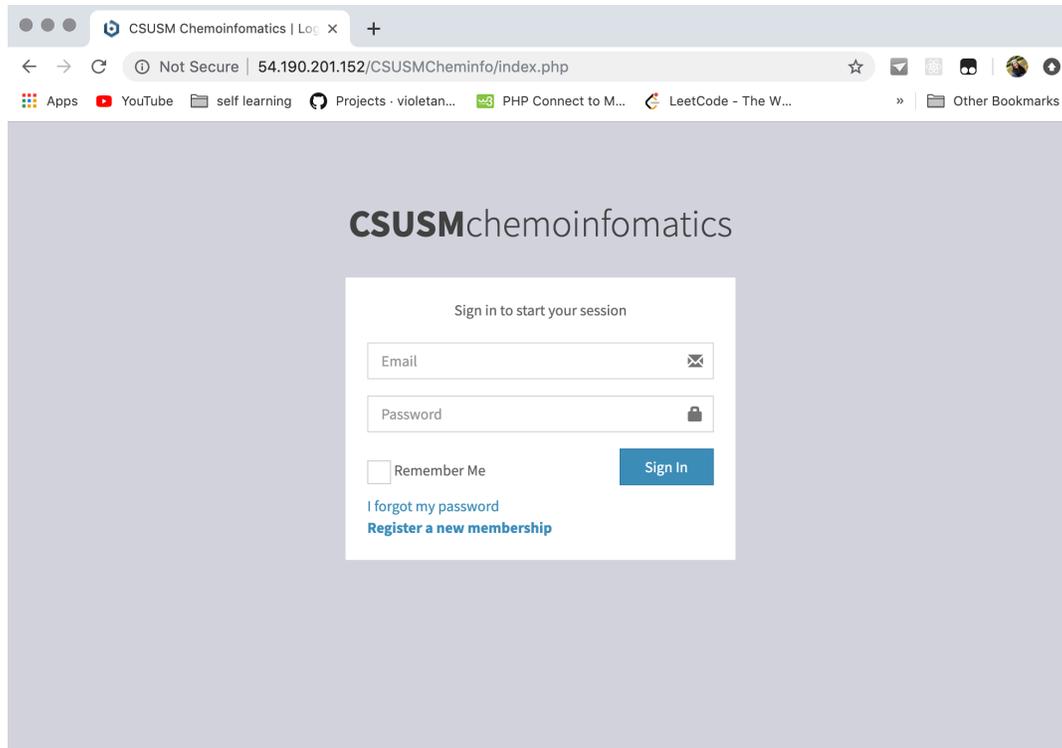


The screenshot shows the same password change form as above. The red error banner now displays the text "Error changing password The passwords don't match." with a close button (X) on the right. The form fields and the "Change the Password" button remain the same.

3.40 change password error- not match

3.2.6 Logout

If a user wants to log out, the user can click the "Logout" button in the navigation bar. Then it will log out and automatically goes to the login page (Graph 3.41).



graph 3.41 log out will leads to sign in page

4. APIs

4.1 getJob

This getJob API is developed for data-minig program to get new job information. The data-mining program could call this getJob API periodically to get the information of the new job created by the user. Datamining program then uses the information to start a test.

The information includes: “status”, represents the status of the job in the database; “message” is the response message, means the API call success or not; “jobId” is the ID of the job in the database; “queue” is the queue value of the job in the database; “jobUserId” is the userId of the job in the database; “userName” is the user name of the user who creates the job in the database; “payload” carries the information about model type and algorithm type; “attempts” represents the job attempts in the database; “data_link” is the address where the data is saved.

Method: GET /api/getJob.php?num=###&token=#####

Content-Type: application/Json

Response Class (Status 200)

An example of the result of getting a new job:

```
[
  {
    "status": 1,
    "message": "OK",
    "jobId": "57",
    "queue": "0",
    "jobUserId": "1",
    "userName": "Jiali Tang",
    "payload": "-mlmType SVM -algorithm GA",
    "attempts": "0",
    "data_link": "/datazip/57-package.zip"
  }
]
```

4.2 updateJob

updateJob is used to update job information in the database. Datamining program can call this API to tell the system to update the job information.

The required attributions are: "status", connection status; "attempts" record how many times the job was tested; "id" is the job id; "reason" explains the testing or errors.

The response attributions are: "status", represents the connection status; "message" implies if the connection is a success or not.

Method: POST: /api/postJob.php?token=#####

Content-Type: application/Json

Response Class (Status 200)

An Example of calling updateJob:

```
{
  "status": 1,
  "attempts": 1,
  "id": 57,
  "reason": "data file incorrect"
}
```

An example of response:

```
{
  "status": 0,
  "message": "OK"
}
```

4.3 postOutput

updateJob is used to update the job output in the database. Datamining program can call this API to tell the system to save the job output.

The required attributions are: “token”, is used to validate the identity; “jobId” which is the id of the job in the database; “userId” which is the id of the user who creates this job.

The response attributions are: “status”, represents the connection status; “message” implies if the connection is a success or not.

Method: POST /api/postOutput.php

Content-Type: application/Json

Response Class (Status 200)

An example of calling postOutput:

```
{
  "token": 123456,
  "jobId": 1,
  "userId": 57
}
```

An example of response:

```
{
  "status": 0,
  "message": "Access not authorized"
}
```

5. Future Work

5.1 Enable more data models

In the next phase of the implementation, none-linear models such as none-linear SVM, and Artificial Neural Network (ANN) [21], and classification data model such as Random Forest (RF) [22] can be added to the implementation part.

5.2 Develop the admin system

Develop an admin system to assist the user with troubleshooting. The admin can check users' test status and execute log. Admin could also add/edit the algorithm/model/disease/target of the database.

Now, this system requires users to upload filtered data to run a data mining test because Dragon doesn't support API. So, an admin system helps manually help users to filter their raw data by download users data file and send E-dragon[30] to calculate under the user's permission.

5.3 Connect with Dragon directly

In the future, if Dragon support APIs, we can develop APIs to connect to it and automatically filter users' data file. This adds to security and privacy.

5.4 Add abstraction layer to support different data mining programs

In this system, we only introduced one type of data-mining program. But in order to support other similar data mining schemes, we can develop an abstraction layer to communicate with them. This makes the system more scalable.

5.5 Improve the cloud infrastructure

To improve the cloud infrastructure, there are valuable tools and services that would improve, speed, scalability, and reliability of the system. Right now, the system is relying on a single instance of elastic compute cloud (EC2), to improve the three characteristics listed before, there are several services that could be added to the stack. The first step will be the creation of a launch template, which will contain the necessary information and configuration such as network requirements, instance type, as well as images used in which the new instances are going to rely on. With a template, the creation of a scaling group will allow setting dynamic parameters, such as CPU load or memory usage which can trigger the creation and invocation of new instances, as well as the termination of those when the demand declines, not only optimizing scalability but also reducing costs. Finally, the implementation of an Elastic Load Balancer will allow the incoming traffic and task request to be distributed among the available

servers (instances) by allocating them to the highest availability one significantly improving the speed and reliability

6. Related Work

The work of Zhong et al. [24] concentrates on the role of Artificial Intelligence (AI) on Drug Discovery. AI and Computer Aided Drug Design (AIDD) techniques have been successfully applied in some stages of the drug discovery, and development pipeline to speed up the process of research and reduce the cost and risk related to preclinical and clinical trials. However, according to Zhong, it is still challenging to provide a physical explanation of the AI-based models. Our modeling techniques do not have any AI flavor, it is simply based on Quantitative Structures Analysis Regression (QSAR) modeling.

Another work is done by Varsou. [25]. Their work presents, with the aid of several representative case studies from drug discovery and computational toxicology, a new cheminformatics platform, called Enalos Suite, that was developed with open source and freely available software. Enalos Suite (<http://enalossuite.novamechanics.com/>) was designed and developed as a tool to address a variety of cheminformatics problems, given that it expedites tasks performed in predictive modeling and allows access, data mining and manipulation for multiple chemical databases (PubChem, UniChem, etc.). Enalos Suite was designed to permit its extension and adjustment to the special field of interest of each user, including, for instance, nanoinformatics, biomedical, and other applications. One of the major difference in our work with Varsou's work is that we have also used Genetic Evolutionary techniques to enhance the training of the models. Further, their work connects to some existing database where our work has its own database and it can be enhanced to be connected to other well-known cheminformatics databases in future work

Other works include Data Mining and Computational Modeling of High-Throughput Screening Datasets. Written by Ekins [26], Web-based drug repurposing tools, written by Sam E. and Athri P [27], DRUG Discovery Using Data Mining written by Charanpreet Kaur and Shweta Bhardwaj [28], and Using Genetic Algorithms for Data Mining Optimization in an Educational Web-Based System written by Behrouz Minaei-Bidgoli et al. [29]. To the best of knowledge, we have not found any paper that offers data mining using QSAR linear, none-linear and classified models with Genetic Evolution algorithms. Most of the papers we have mentioned in the references provide part of the work and possibly other functions but none provide a strong database as a backend to support the data as we offer in our work.

7. Reference

1. Gene M. Ko, A. Srinivas Reddy, Rajni Garg, Sunil Kumar, Ahmad R. Hadaegh, "Computational Modeling Methods for QSAR Studies on HIV-1 Integrase Inhibitors (2005-2010)," [Curr Comput Aided Drug Des.](#) 2012; 8(4):255-70 (ISSN: 1875-6697). 2012.
2. Thakor F, Hadaegh A, Zhang X. Comparative study of Differential Evolutionary-Binary Particle Swarm Optimization (DE-BPSO) algorithm as a feature selection technique with different linear regression models for analysis of HIV-1 Integrase Inhibition features of Aryl β -Diketo Acids. [Proceedings of 9th International Conference on Bioinformatics and Computational Biology \(BICOB 2017\) Honolulu, Hawaii](#), USA, ISBN: 978-1-943436-07-1. pages 179-184 March 22, 2017.
3. Kane I, and Hadaegh A. R. Non-linear Quantitative Structure-Activity Relationship (QSAR) Models for the Prediction of HIV Drug Performance. 24th International Conference on Software Engineering and Data Engineering (SEDE-2015). Page 63-68. Vol 1, ISBN: 9781510812277, San Diego, CA. October 2015.
4. Galvan R, Kashani, M and Hadaegh A. R, Improving Pharmacological Research of HIV-1 Integrase Inhibition Using Differential Evolution-Binary Particle Swarm Optimization and Non-Linear Adaptive Boosting Random Forest Regression. 2015 IEEE International Workshop on Data Integration and Mining San Francisco. Information Reuse and Integration (IRI), 2015 IEEE International Conference on.. Pages: 485-490. DOI: [10.1109/IRI.2015.80](#). INSPEC Accession Number: 15556631. San Francisco, CA. August 2015.
5. Kashani, M, Galvan R, and Hadaegh A. R, Improving the Feature Selection for the Development of Linear Model for Discovery of HIV-1 Integrase Inhibitors. ABDA'15 International Conference on Advances in Big Data Analytics. [In Proceeding of the 2015 International Conferences On Advances On Big Data Analyses](#). Pages 150-154. ISBN: 1-60132-411-1. Las Vegas, Nevada. July 2015.
6. Gene M. Ko, Rajni Garg, Sunil Kumar, Barbara A. Bailey, Ahmad R. Hadaegh, "A Hybridized Evolutionary Algorithm for Feature Selection of Chemical Descriptors for Computational QSAR Modeling of HIV-1 Integrase Inhibitors," Computational Science Curriculum Development Forum and Applied Computational Science and Engineering Student Support for Industry, San Diego State University, March 1, 2013.
7. Gene M. Ko, Rajni Garg, Sunil Kumar, Barbara A. Bailey, Ahmad R. Hadaegh, "Differential Evolution-Binary Particle Swarm Optimization for the Analysis of Aryl β -Kiketo Acids for HIV-1

- Integres Inhibition. WCCI 2012 IEEE World Congress on Computational Intelligence. Brisbane Australia, pp. 1849-1855. June, 2012
8. Gene M. Ko, A. Srinivas Reddy, Sunil Kumar, Barbara A. Bailey, Rajni Garg, Ahmad R. Hadaegh, "Evolutionary Computational Modeling of β -Diketo Acids for Virtual Screening of HIV-1 Integrase Inhibitors", IEEE World Congress on Computational Intelligence, Brisbane, Australia, June 10-15, 2012.
 9. Gene M. Ko, A. Srinivas Reddy, Sunil Kumar, Barbara A. Bailey, Rajni Garg, Ahmad R. Hadaegh, "Evolutionary Computational Modeling of β -Diketo Acids for Virtual Screening of HIV-1 Integrase Inhibitors," IEEE World Congress on Computational Intelligence, Brisbane, Australia, June 10-15, 2012.
 10. Gene M. Ko, A. Srinivas Reddy, Rajni Garg, Sunil Kumar, Ahmad R. Hadaegh, "Evolutionary Computational Modeling of β -Diketo Acids for Virtual Screening of HIV-1 Integrase Inhibitors," 243rd National Meeting of the American Chemical Society, San Diego, CA, March 25-29, 2012.
 11. Chris L. Turner, Miguel A. Gonzales, Gene M. Ko, Ahmad R. Hadaegh, "Genetic Algorithm Modeling of Dimeric Aryl Diketo Acid Inhibitors for HIV-1 Integrase," 243rd National Meeting of the American Chemical Society, San Diego, CA, March 25-29, 2012.
 12. Miguel A. Gonzales, Chris L. Turner, Gene M. Ko, Ahmad R. Hadaegh, "Binary Particle Swarm Optimization Model of Dimeric Aryl Diketo Acid Inhibitors for HIV-1 Integrase," 243rd National Meeting of the American Chemical Society, San Diego, CA, March 25-29, 2012.
 13. Gene M. Ko, A. Srinivas Reddy, Sunil Kumar, Rajni Garg, Ahmad R. Hadaegh, "Analysis of HIV-1 Integrase Inhibitors Using Computational QSAR Modeling," Computational Science Curriculum Development Forum and Applied Computational Science and Engineering Student Support for Industry, San Diego State University, March 23, 2012 . <Awarded Los Alamos Award for Outstanding Presentation>.
 14. Gene M. Ko, Chris L. Turner, Miguel A. Gonzales, Ahmad R. Hadaegh, "QSAR Analysis of HIV-1 Integrase Inhibitors Using Binary Particle Swarm Optimization," 43rd Western Regional Meeting of the American Chemical Society, Pasadena, CA, November 10-12, 2011.
 15. Gene M. Ko, Chris L. Turner, Miguel A. Gonzales, Ahmad R. Hadaegh. "QSAR Study on Dimeric Aryl Diketo Acid HIV-1 Integrase Inhibition Using Particle Swarm Optimization and Multiple Linear Regression," National Biomedical Computing Resource Summer Institute, University of California, San Diego, August 1-5, 2011

16. Zhang X, Bolanos L, Callow G, Hadaegh A. R, and Read B. Resequencing the Genome of Different *E. huxleyi* Isolates. Plant and Animal Genome XVII Conference, January 10-14, 2009
17. Garg R, Reddy S. A, Zhang X, and Hadeagh A. R. MUT-HIV: Mutation database of HIV proteases, American Chemical Society (ACS) 234th National Meeting & Exposition August 19-23, 2007, Boston, MA USA CINF 42
18. MLR: <http://www.stat.yale.edu/Courses/1997-98/101/linmult.htm>
19. PLSR: <https://www.mathworks.com/help/stats/plsregress.html>
20. SVM: https://en.wikipedia.org/wiki/Support_vector_machine
21. ANN: https://en.wikipedia.org/wiki/Artificial_neural_network
22. RF: https://en.wikipedia.org/wiki/Random_forest
23. <http://www.rosebt.com/blog/predictive-descriptive-prescriptive-analytics>
24. Zhong et al. Artificial intelligence in drug design. *Sci China Life Sci.* 2018 Jul 18. doi: 10.1007/s11427-018-9342-2. [Epub ahead of print]
25. [Varsou DD1](#), [Nikolakopoulos S1](#), [Tsoumanis A1](#), [Melagraki G2](#), [Afantitis A3](#). Enalos Suite: New Cheminformatics Platform for Drug Discovery and Computational Toxicology. *Methods Mol Biol.* 2018; 1800:287-311. doi: 10.1007/978-1-4939-7899-1_14. 2018
26. [Ekins S](#), et al.. Data Mining and Computational Modeling of High-Throughput Screening Datasets. *Methods Mol Biol.* 2018; 1755:197-221. doi: 10.1007/978-1-4939-7724-6_14.
27. [Sam E](#), [Athri P](#). Web-based drug repurposing tools: a survey. *Brief Bioinform.* 2017 Oct 6. doi: 10.1093/bib/bbx125. [Epub ahead of print]
28. Charanpreet Kaur and Shweta Bhardwaj, DRUG Discovery Using Data Mining International Journal of Information and Computation Technology. ISSN 0974-2239 Volume 4, Number 4 (2014), pp. 335-342 © International Research Publications House [http://www. irphouse.com /ijict.htm](http://www.irphouse.com/ijict.htm)
29. Behrouz Minaei-Bidgoli and William F. Punch, Using Genetic Algorithms for Data Mining Optimization in an Educational Web-Based System, Genetic Algorithms Research and Applications Group (GARAGE) Department of Computer Science & Engineering Michigan State University 2340 Engineering Building East Lansing, MI 48824 {minaeibi,punch}@cse.msu.edu, 2003.
30. https://chm.kode-solutions.net/products_dragon.php
31. GA: https://en.wikipedia.org/wiki/Genetic_algorithm
32. DE: https://en.wikipedia.org/wiki/Differential_evolution

33. QSAR: https://en.wikipedia.org/wiki/Quantitative_structure%E2%80%93activity_relationship
34. AWS LightSail: https://aws.amazon.com/lightsail/?nc2=h_ql_prod_fs_ls
35. AWS EC2 Server: <https://aws.amazon.com/ec2/>