

1. Home	3
1.1 Requirements	6
1.1.1 Background	7
1.1.2 Functional Requirements	9
1.1.3 Motivational Model	10
1.1.4 Non-Functional Requirements	11
1.1.5 Paper Prototype	12
1.1.6 Personas	13
1.1.7 Product Backlog	15
1.1.8 User Stories	17
1.2 Meetings	19
1.2.1 Client Meetings	20
1.2.1.1 10/03/2023 Client Meeting	21
1.2.1.2 23/03/2023 Client Meeting	22
1.2.1.3 03/04/2023 Client Meeting	23
1.2.1.4 17/04/2023 Client Meeting	24
1.2.1.5 01/05/2023 Client Meeting	25
1.2.1.6 08/05/2023 Client Meeting	26
1.2.1.7 15/05/2023 Client Meeting	27
1.2.1.8 22/05/2023 Client Meeting	28
1.2.2 Supervisor Meetings	29
1.2.2.1 09/03/2023 Meeting notes	30
1.2.2.2 17/03/2023 Meeting notes	31
1.2.2.3 30/03/2023 Meeting notes	32
1.2.2.4 07/04/2023 Meeting notes	33
1.2.2.5 20/04/2023 Meeting notes	34
1.2.2.6 27/04/2023 Meeting notes	35
1.2.2.7 12/05/2023 Meeting notes	36
1.2.2.8 18/05/2023 Meeting notes	37
1.2.2.9 25/05/2023 Meeting notes	38
1.2.3 Teams Meetings	39
1.2.3.1 13/03/2023 Team Meeting	40
1.2.3.2 15/03/2023 Team Meeting	41
1.2.3.3 17/03/2023 Team Meeting	42
1.2.3.4 22/03/2023 Team meeting	43
1.2.3.5 26/03/2023 Team meeting	44
1.2.3.6 29/03/2023 Team Meeting	45
1.2.3.7 19/04/2023 Team Meeting	46
1.2.3.8 26/04/2023 Team Meeting	47
1.2.3.9 25/05/2023 Team Meeting	48
1.2.3.10 07/06/2023 Team Meeting	49
1.3 Development	50
1.3.1 Guide	51
1.3.2 Development Environment	52
1.3.3 Deployment Plan	55
1.3.4 Developing Milestones	58
1.3.5 Technical Analysis	63
1.3.6 GitHub Version Control	65
1.3.7 Mapping System	66
1.3.7.1 Mapping algorithm	67
1.3.7.2 Mapping analysis	70
1.3.7.3 Mapping Scenario	75
1.4 Sprint Plan & Review	76
1.4.1 Kanban in Trello	77
1.4.1.1 Sprint1	78
1.4.1.2 Sprint2	79
1.4.1.3 Sprint3	80
1.4.1.4 Sprint4	81
1.4.2 Sprint Plan	82
1.4.2.1 Sprint 2 Plan	83
1.4.2.2 Sprint 3 Plan	85
1.4.2.3 Sprint 4 Plan	87
1.4.3 Sprint Backlog	89
1.4.3.1 Sprint 2 Backlog	90
1.4.3.2 Sprint 3 Backlog	92
1.4.3.3 Sprint 4 Backlog	93
1.4.4 Sprint Review	94
1.4.4.1 Sprint2 - Review	95
1.4.4.2 Sprint3 - Review	97
1.4.4.3 Sprint4 - Review	99
1.4.5 Technology Plan	100
1.5 Quality	101
1.5.1 Code Review	102
1.5.1.1 Algorithm Review	103
1.5.1.2 Back-end Review	107
1.5.1.3 ChatGPT-assisted Code Review	114
1.5.1.4 Front-end review	117
1.5.2 Testing & Evaluation	131
1.5.2.1 Test Case	132

1.5.3 Code review checklist	135
1.5.4 Coding Standards	136
1.5.5 Risk Management	137
1.6 Product	138
1.7 Presentations	139
1.8 Research	149
1.8.1 Research items	150
2. Project Details	151

Home



DI

Digital Health (DI)

COMP90082-2023-SM1-DI-Redback

Project Details Trello

Quick Access

Process Documentation

Product Backlog	Meetings	Sprint Plan & Review	Development
Trello	Research		

Product Documentation

Quality	Risk Management	Sprint Plan & Review	Requirements
Mapping System	Video	Presentations	

Roadmap And Assignments Deadline

- Client Meeting Every Friday 4pm – 5pm
- Sprint 1: Inception (or design sprint) 2023-3-24 13:59
- Peer Review #1 (self+team) [10-15mins assessment task] 2023-4-6 23:59
- Sprint 2: Development 2023-5-1 23:00
- Peer Review #2 (self+team) [10-15mins assessment task] 2023-5-19 23:59
- Sprint 3: Development 2023-5-26 13:00
- Final presentation 2023-5-26 23:59
- Professional Skills #2 (self-assessment) [10mins assessment task] 2023-5-26 23:59
- Contributions to project [no submission here] 2023-6-9 13:00
- Sprint 4: Product 2023-6-9 13:00
- Self-reflection report[1-2days assessment task] 2023-6-9 13:00

Stakeholders

Name	Role	Contact
Mauro Mello Jr	Supervisor	Email: mauro.mellojr@unimelb.edu.au

Dr Vlada Rozova	Client	Email: vlada.rozova@unimelb.edu.au
Dr Mike Conway	Client	Email: mike.conway@unimelb.edu.au
Dr Daniel Capurro	Client	Email: dcapurro@unimelb.edu.au

Team

Name	Role	Photo	Contact
Lang Cao	Product Owner	 A head-and-shoulders portrait of a young man with short dark hair, wearing a black t-shirt.	Email: lancao@student.unimelb.edu.au Phone: 0412856668
Mingjun GAO	Developer	 A head-and-shoulders portrait of a young man with short dark hair, wearing a blue jacket over a grey shirt.	Email: mingjung1@student.unimelb.edu.au Phone: 0414640105

Jiacheng WANG	Developer		Email: jiacwang3@student.unimelb.edu.au Phone: 0481275596
Yanxi KE	Developer		Email: yake@student.unimelb.edu.au Phone: 0424641265
Runzhe Hua	Scrum Master		Email: runzheh@student.unimelb.edu.au Phone: 0432959598

Recent space activity

- Runzhe HUA
 - Sprint 4 Plan 44 •
 - Sprint 3 Plan 44 •
 - Sprint4 - Review 1 •
 - Teams Meetings 1 •
 - Supervisor Meetings 1 •

Space contributors

- Runzhe HUA (44)
- Jiacheng WANG (1)
- Yanxi KE (1)
- Mingjun GAO (5)
- Lang Cao (5)
- ...

Requirements

- Background
- Functional Requirements
- Non-Functional Requirements
- Motivational Model
- Paper Prototype
- Personas
- User Stories

Background

- Background
- About Clients
- Client Goals
- Scope

Background

When a patient sees a doctor, the doctor records the patient's symptoms. However, doctors will abbreviate or modify it for the convenience of recording according to personal habits. The description of the symptoms that led to these symptoms is not part of the Universal Indication List. This makes it difficult to count and analyze symptoms. Therefore, turning the original file into a Universal Indication List is necessary.

About Clients

There are three industry partners becoming clients. there are Vlada Rozova, Mike Conway, and Daniel Capurro. The first two of them will be our contacts.

Client Goals

The core functionality of the project is to allow users to upload the txt files, algorithmically map the raw data files (txt) to a list of the Universal Indication List, and host this functionality through a local interface. The customer wishes to implement the system in two modes. One mode is for researchers to directly upload data and then download transformed data from the system. Another mode is that after the researchers upload the data, they modify the wrongly mapped data through the interface to improve the system algorithm. In order to ensure that only relevant researchers can use the system, user login and registration are required. But only a handful of researchers will use the system, so it requires no administrators. In addition, the result could be visualisation by the mapping performance, including the number that can be mapped and cannot be mapped, and mapping based on which tools. Each mapping will be recorded and the user also can see their mapping history through their account.

Word Explanation

UIL: full name is the Universal Indication List, which is provided by clients and will be updated in the future.

SNOMED CT: we are using SNOMED CT-AU in this project. It is a database containing lots of words. More information will introduction on the development page.

Docker: Docker Desktop is the software used for connecting to SNOMED CT.

Raw text: Provided by clients, an example of input, each line should map to one in UIL

System

According to the customer's goals, this mapping system is based on the website (In localhost). Therefore, this project has two central systems: a mapping system and an account management system. The mapping system includes the function mapping raw text to the Universal Indication List. The account management system includes recording user information to log in and find the history mapping system.

Scope

In Scope

Digit Health aims to provide the University of Melbourne researchers with a mapping system so that they can map raw plaintext to a list of common instructions. Web-based and natural language processing technologies are necessary. Mapping algorithms should be able to learn and improve over time based on user feedback. We will implement the system within four months.

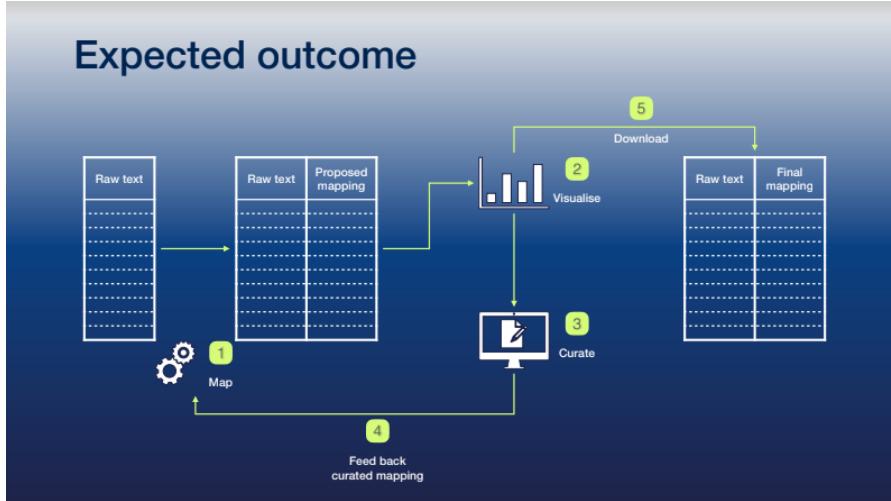
Out of Scope

This is also available for other business needs. If the negotiations are successful in the future, relevant interfaces may be provided for external personnel (Hospitals, other university medical research centres, and national medical departments).

Time Line



Functional Requirements



This is the figure provided by clients, including the 5-main function.

Version 3

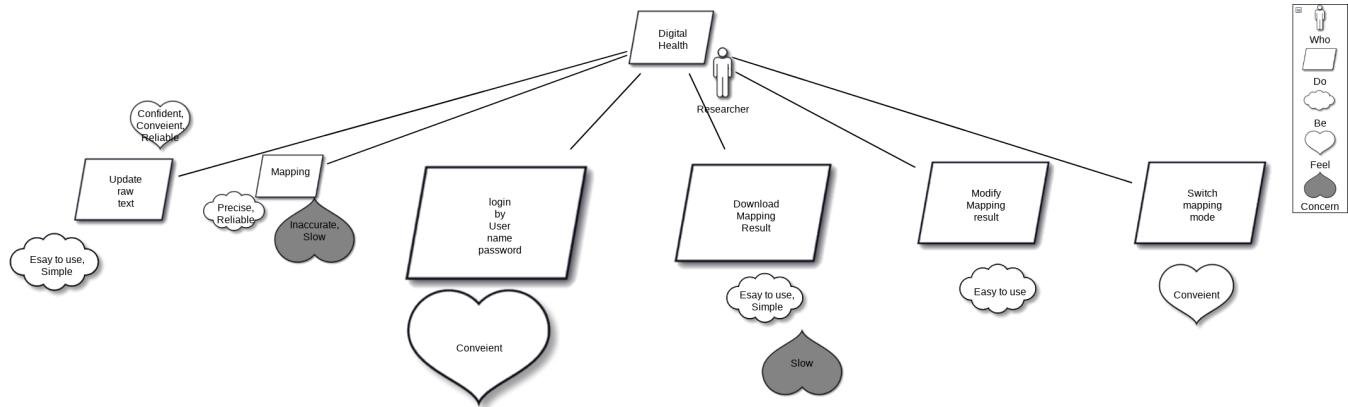
- 1. Mapping system**
 - The system should be able to convert non-standard disease descriptions into the Universal Indication List.
 - There are three columns in the mapping result, raw text, mapping result, mapping based on SNOMED CT or UIL.
 - The result can be the mapping result or Non-Match.
 - Ability to submit multiple files at once.
 - Upon uploading files, the system will indicate the state of the process(processing or finish).
 - The system should allow researchers to modify the results.
 - It should show in different colours if the result has been modified.
 - Once the mapping result has been modified, the raw text and new mapping result will be saved as a new dictionary in the local file.
 - Users can only input data that are within SNOMED-CT and UIL as possible modification values.
 - The system should allow researchers to export the results into a file
 - The file should be a CSV file to download.
 - The system should visualize the accuracy of the mapping result.
 - The mapping result should display how many can be mapped and how many Non-Match
 - The mapping result in the Universal Indication List is then flagged as UIL. Otherwise, flag it as SNOMED CT if the result can be found in SNOMED CT but cannot find it in UIL.
 - The mapping result will also indicate the distance between the mapping result and the raw data, as well as the distances between other potential results and the raw data.
 - The system can be switched to pipeline mode
 - Others can call the system's interface directly, upload data, and obtain mapping results in CSV format.
- 2. Account system**
 - The system can register and log in to the user.
 - Email, name and password are recorded in the system.
 - The Email will contain the random password to log in the first time and then they can set the password by themselves.
 - The user can view all the mapping history.
 - The system keeps a record of all mapping history.
 - The user can look up the mapping history of all files uploaded by all users.

Motivational Model

Do-Be-Feel-Who List

Who	Do	Be	Feel
• Researcher	<ul style="list-style-type: none"> • Login by username and password • Update raw text • Download Mapping Result • Modify Mapping result • Switch mapping mode • Mapping 	<ul style="list-style-type: none"> • Easy to use • Simple • Responsive • Simple • Intuitive • Precise 	<ul style="list-style-type: none"> • Confident • In control • Convenient • Reliable

Motivational Model



Non-Functional Requirements

Version 3

1. The system needs to map in a limited time.
 - a. If multiple users are mapping the data at the same time, the system needs to make sure all of them are mapping efficiently.
2. High mapping accuracy is necessary.
 - a. Mapping as much as the system can and less error.
3. the system installation needs to be simple and fast
 - a. it needs to use the docker technology to install
 - b. The installation needs to consider the system and environment of the local server in order to install different versions.
4. Keep the information safety in the service.
 - a. Network security needs to be ensured.

Paper Prototype

V2.0:

This is the final version that we used in the development

what we do:

- According to customer needs, all pages related to the administrator role have been deleted.
- At the same time, the functions related to the manage account page are also deleted. The customer is very satisfied with the remaining prototype and said that it should now focus on the prototype verification of the main function mapping system.

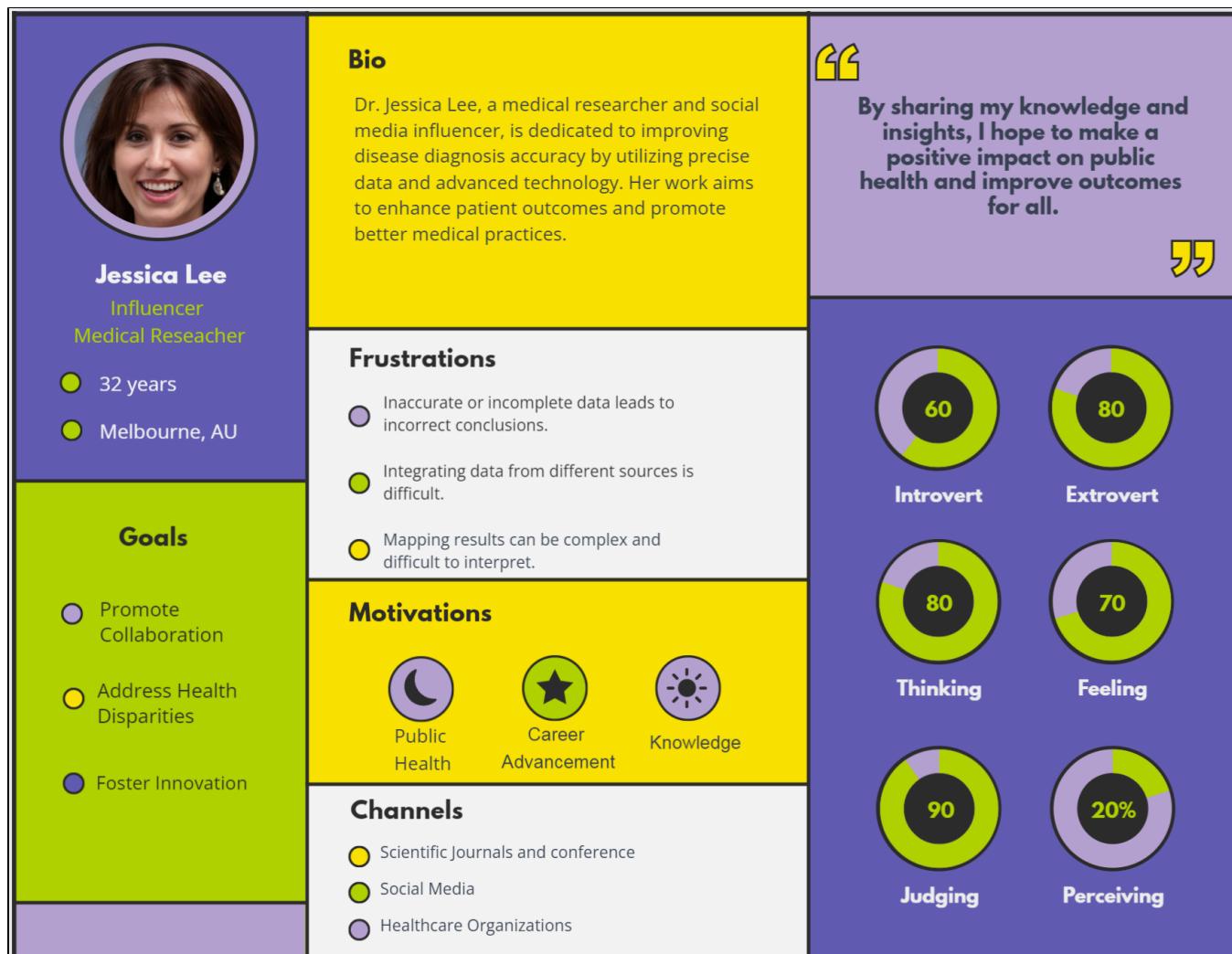
Login page	Register Page	Mapping Page								
 <p>Login</p> <p>User Name admin1</p> <p>Password Please Enter Password</p> <p><input checked="" type="checkbox"/> Remember Me</p> <p>Login</p> <p>Don't have an account? Sign Up</p>	 <p>Register</p> <p>* Username <input type="text"/></p> <p>* Email <input type="text"/></p> <p>* Password <input type="password"/></p> <p>* Confirm Password <input type="password"/></p> <p>Register</p>	 <p>Mapping System Upload Mappings</p> <table border="1"><thead><tr><th>ID</th><th>Title</th><th>Description</th><th>Created At</th></tr></thead><tbody><tr><td>f660b2a4-8469-4544-a3ae-f6f167ffcc7f</td><td></td><td></td><td>2023-04-16 22:55:58</td></tr></tbody></table>	ID	Title	Description	Created At	f660b2a4-8469-4544-a3ae-f6f167ffcc7f			2023-04-16 22:55:58
ID	Title	Description	Created At							
f660b2a4-8469-4544-a3ae-f6f167ffcc7f			2023-04-16 22:55:58							

Personas

Persona 1

Name: Jessica Lee

Occupation: Medical Researcher



How did the Frustrations and characteristics of Jessica Lee influence the system Design?

Her frustrations:

- Accurate data quality: Provide data quality checking tools in the software to ensure the accuracy and integrity of the data. (e.g. Automated Data Cleaning Functionality)
- Multi-source data integration: Data merging, data format conversion function. (e.g. merging multiple JSON files with the same structure or converting JSON files to csv)
- Easy-to-interpret mapping results: Design a visual, interactive mapping tool to help healthcare professionals understand and interpret mapping results better. (e.g. An easy-to-understand user interface that can show various mapping results, providing tools to help users interpret the results).

Jessica's characteristics:

- Providing tools that help with decision-making and analysis when designing your software. (e.g. interactive analysis tool helps users analyze and compare different data sets)
- Since her personality focuses on judgment and decision-making, the design could consider giving me neutral advice based on previous mapping results or building an intelligent search engine to help her access
- A structured workflow that allows medical staff to follow a predefined process for data mapping and analysis to improve efficiency and accuracy

Product Backlog

Document Status	Version	Date	Comment
DONE	2.0	2023-3-23	Modifying requirements according to the results of the client meeting #2.
DONE	1.1	2023-3-21	Requirements plan
DONE	1.0	2023-3-19	Requirements plan (Requirements draft)

V2.0

User story link: [User Stories](#)

Sprint2 Backlog: [Sprint 2 Backlog](#)

- System Environment

- US17-Need to use docker for software deployment on the server

- Mapping System

- US08-Web Interface with a nav bar

- nav bar has "mapping", and "upload", the right side is the account state and "logout" button.

- US05-The user can upload csv, txt files to the system.
 - US14-Ability to submit multiple files at once.
 - US12-it will show each upload files' state(processing or finish)
 - US01-The system should be able to convert non-standard disease descriptions into the Universal Indication List.

- There are three columns in the mapping result, raw text, mapping result, mapping based on SNOMED CT or UIL
 - The result can be the mapping result or Non-Match.
 - Upon uploading files, the system will indicate the state of the process(processing or finish).

- US04-The system should allow researchers to modify the results.

- It should show in different colors if the result has been modified.
 - Once the mapping result has been modified, the raw text and new mapping result will be saved as a new dictionary in the local file.
 - US11-Users can only input data that are within SNOMED-CT and UIL as possible modification values.

- US02-The system should allow researchers to export the results into a file

- The file should be a CSV file to download.

- US15-The system should show the mapping result.

- US10-The mapping result will also indicate the distance between the mapping result and the raw data, as well as the distances between other potential results and the raw data.

- US03-The system should visualize the accuracy of the mapping result.

- The mapping result should display how many can be mapped and how many Non-Match
 - The mapping result in the Universal Indication List is then flagged as UIL. Otherwise, flag it as SNOMED CT if the result can be found in SNOMED CT but cannot find it in UIL.

- US06-The system can be switched to pipeline mode

- Others can call the system's interface directly, upload data, and obtain mapping results in CSV format.

- Account System

- US07-The system can register and log in to the user.

- Email, name and password are recorded in the system.

- The Email will contain the random password to log in the first time and then they can set the password by themselves.

- US16-The user has a profile page that contains personal information
- The user can view all the mapping history.

- The system keeps a record of all mapping history.
- US13-The user can look up the mapping history of all files uploaded by all users.
- US09-The user can modify other user's modify

User Stories

Version

Version ID	Description	Date
3.0	After the fourth clients meeting. The client require some new function. Show the result distancefile processing state, modify limit.	17 /04/2023
2.1	This is based on the third clients meeting, having some adustion according on progress we showing ,and add the dashboard user stories.	03/04 /2023
2.0	Following the second client meeting, we have gained a comprehensive understanding of the client's requirements. A noteworthy change that has been identified pertains to the discontinuation of the admin user.	23/03 /2023
1.0	After the first client meeting. We initialise the user stories base on the meeting and analysis.	18/03 /2023

V 3.0 User Story

StoryID	User	Story	Priority	DOD (Definition of Done)	Story points
1	Resear cher	As a researcher, I want to view the list of standardized disease descriptions generated by the system so that I can take future action.	Must have	The mapping result is correct and mapping time is a limitation	3
2	Resear cher	As a researcher, I want to download the mapping result so that I can use the mapping result for other research or save them in local computer.	Must have	The mapping result can be downloaded as a local CSV file	5
3	Resear cher	As a researcher, I want to visualize the mapping results so that I can analyze the accuracy and reliability of the system or knowhow much of the data entered is valid.	Must have	The researcher can see the number of Non-match and Match results.	5
4	Resear cher	As a researcher, I want to modify mapping results so that I can help the system with mapping results and improve the system algorithm.	Must have	If the mapping result modifies by the researcher, it will be saved and then if the raw text comes again it will map to the modified result.	4
5	Resear cher	As a researcher, I want to upload Raw data to the mapping system so that I can use the system to get the mapping results.	Must have	There is a button or box that allows the user to upload a file and store it as temporary data.	5
6	Resear cher	As a researcher, I want to switch the system mode so that mapping large raw data like a pipeline.	Must have	The researcher can upload multiple files, then process them and download them as conveniently as possible without modifying the function or display table.	3
7	Resear cher	As a researcher, I want to have my own username and password so that log in to the system.	Must have	Each researcher has their own Email as username and their own password to log in.	4
8	Resear cher	As a researcher, I want to have a nav bar so that I can switch between uploading and mapping page and can view the user name and sign out.	Must have	All functions for nav bar are implemented.	3
9	Resear cher	As a researcher, I want to modify other researchers' modifications in the dashboard.	Must have	The research can modify other reseracher's upload and modify in the dashboard.	4
10	Resear cher	As a researcher, I want to see the confidence of the mapping result and other potential results when I see the result.	Must have	The mapping result should also indicate the distance between the mapping result and the raw text and the distances between the raw text and other possible describe in SNOMED-CT.	3
11	Resear cher	As a researcher, I want to see if I entered the right word in SNOMED-CT when modifying the results.	Must have	Researcher can only input the available word when they modify the result.	3
12	Resear cher	As a researcher, I want to see the state of uploaded files (processing , finished).	Must have	The upload files will show the state(processing or finish) in the dashboard.	3
13	Resear cher	As a researcher, I want to see each uploaded processed files' visualization in the dashboard.	Must have	The dashboard will show the each processed files' visualization.	5
14	Resear cher	As a researcher, I want to upload many unprocessed files at the same time.	Should have	The research can upload 10 unprocessed files together.	3
15	Resear cher	As a researcher, I want to see the mapping results so that I can analyze the system's reliability and the system algorithm's defects through my own experience.	Should have	The mapping results shows in a table.	5

16	Researcher	As a researcher, I want to a profile page so that I can modify my own user information.	Should have	The profile page shows user information and can click edit to modify user information and then save.	2
17	Researcher	As a researcher, I want to use docker to deploy this system to the server	Must have	the system can simplyly, easily use docker to deploy this system to the server.	5
18	Researcher	As a researcher, I want to upload the new list of standardized disease descriptions to the system so that I can get the result i wanted.	Must have	The mapping result can be download or give the way to download	3

Meetings

Incomplete tasks from meetings

Meeting categories

Client Meetings	Supervised Team Meetings	Team Meetings
-----------------	--------------------------	---------------

All meeting notes

03/04/2023 Client Meeting	Jiacheng WANG	2
17/04/2023 Client Meeting	Runzhe HUA	01 , 2023
27/04/2023 Meeting notes	Runzhe HUA	01 , 2023
26/03/2023 Team meeting	Runzhe HUA	25 , 2023
20/04/2023 Meeting notes	Runzhe HUA	23 , 2023
23/03/2023 Client Meeting	Jiacheng WANG	16 , 2023
30/03/2023 Meeting notes	Jiacheng WANG	04 , 2023
17/03/2023 Meeting notes	Runzhe HUA	01 , 2023
17/03/2023 Team Meeting	Runzhe HUA	23 , 2023
15/03/2023 Team Meeting	Runzhe HUA	23 , 2023
13/03/2023 Team Meeting	Runzhe HUA	23 , 2023
09/03/2023 Meeting notes	Runzhe HUA	23 , 2023
10/03/2023 Client Meeting	Runzhe HUA	23 , 2023

Client Meetings

Our Clients meeting all face-to-face

- [10/03/2023 Client Meeting](#)
- [23/03/2023 Client Meeting](#)
- [03/04/2023 Client Meeting](#)
- [17/04/2023 Client Meeting](#)
- [01/05/2023 Client Meeting](#)
- [08/05/2023 Client Meeting](#)
- [15/05/2023 Client Meeting](#)
- [22/05/2023 Client Meeting](#)

10/03/2023 Client Meeting

Date

2023-3-10

Attendees

- Daniel Capurro
- Vlada Rozova
- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Get to know the client
- Know the information can be provided by clients
- Understand what the client needs

Discussion items

Item	Notes
Introduction by Daniel Capurro	<ul style="list-style-type: none">• Presentation by Daniel Capurro introduces his pattern and other clients• Expects to achieve• Show the working process for the current project• Show and provide the raw plaintext
Introduction by Vlada Rozova	<ul style="list-style-type: none">• Introduction by herself• Some additional information provided• Mentioned she will be our contact in the future• Mapping accuracy needs to be increased over time.
Ask for requirements	<ul style="list-style-type: none">• This is a web-based project• The computing algorithm needs to be updated by modifying the mapping result• OntoServer account with the license will be given by the university.• the Universal Indication List will be given in the future• We ask for technical details of what needs to be implemented

Action items

- Research whether the data given by the customer can be matched using Ontoserver.

23/03/2023 Client Meeting

Date

2023-3-23

Attendees

- Dr Vlada Rozova (client)
- Dr Mike Conway (client)
- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Review and clarify the client's requirements for the project interface.
- Inform the clients of the technical problems we encounter when conducting medical information retrieval
- Confirm our user stories with clients
- Understand and confirm high-priority requirements from clients
- Confirm the weekly meeting

Discussion items

Item	Notes
Discussing the desired functionality, design, and user experience for the interface.	<ul style="list-style-type: none">• Presented and discussed our current prototype with clients.• The client reiterated their requirements and again elaborated on the system's two modes.
Problems have encountered	<ul style="list-style-type: none">• Clients suggest finding alternative solutions if OntoServer is still unavailable.• We discussed downloading SNOMED CT directly but decided against it due to difficulties in parsing it.• Clients will take a look at the UIL(No elements) problem
Confirm and update user stories	<ul style="list-style-type: none">• The client advised that since only a few researchers will use this system, the admin is no longer needed.• The client pointed out that using OntoServer is not mandatory, and if there is an alternative, it can be used.

Action items

- Investigate alternative solutions to OntoServer, and keep the client updated on any progress or changes.
- Follow up with the client regarding the UIL(No elements) problem and determine the next steps to address it.
- Update the user stories based on the feedback from the client, and remove the admin feature from the system.

03/04/2023 Client Meeting

Date

2023-4-3

Attendees

- Dr Vlada Rozova (client)
- Dr Mike Conway (client)
- Mauro Mello Jr
- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Showing our clients and supervisor the five main functions we have completed.
- Discuss the mapping result if there are multiple options in the UIL.
- Discussion of UI requirements and login functionality details.
- Seeking customers' opinions and feedback on whether we will use the RNN and use a vector to calculate the distance in the future.
- Confirm the next meeting date.

Discussion items

Item	Notes
Client feedback on matching results	<ul style="list-style-type: none">• Clients are satisfied with the raw text can be matched in the normal situation• The client is explicitly unlikely to provide more data that can be used for training• The supervisor has provided access to a medical repository called Transient Global Amnesia (TGA). This presents an opportunity for us to acquire additional healthcare training data beyond raw data.
Client feedback on existing non-algorithmic features	<ul style="list-style-type: none">• The client is satisfied with the overall outcome as it meets their requirements for the essential functionalities they were seeking.• In visualization, display more information<ul style="list-style-type: none">• Matched by SNOMED CT but not in UIL• So, there will be displaying three columns, Matched, Non-Matched and Matched by SNOMED CT.• For the login system.<ul style="list-style-type: none">• Email as user name for the login system.• Sending the random password through Email as initial information for login
Feedback on whether to use RNN algorithms in the future	<ul style="list-style-type: none">• The client is not optimistic about the deployment of RNN, partly due to the lack of data and concerns that it may be overkill for this task.• The client recommended a distance-based machine learning algorithm and emphasized the importance of keeping it lightweight.

Action items

- Login System
- Implementation of basic login functionality.
- Research on Transient Global Amnesia (TGA) to determine its potential to provide additional training data for the project.
- Extension of existing basic functionalities including data visualization and modification.
- Exploration of alternative machine learning algorithms for data matching.

17/04/2023 Client Meeting

Date

2023-4-17

Attendees

- Dr Vlada Rozova (client)
- Dr Mike Conway (client)
- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Demonstrate the new login and register pages.
- Demonstrate the new layout of functions.
- Explain the RNN we implemented.
- Present the logical map of our mapping system overall.
- Seeking customers' opinions and feedback

Discussion items

Item	Notes
Demonstration of new Layout	<ul style="list-style-type: none">• Customers are generally satisfied with the new layout
Discussion on technical details about the RNN	<ul style="list-style-type: none">• How to handle some abnormal data• The client thinks we should try the distance-based function, such as edited distance. As it is much faster and simpler.
Discussion on the result of RNN	<ul style="list-style-type: none">• The client emphasizes that the mapping result that is certain by UIL could be better since it's more detailed and specific• The client thinks we need an easier method to sort the non-match result, so they can easily access all the non-match results.
Future improvements	<ul style="list-style-type: none">• Need to indicate the status of files• Need tracking the history of changes and who made the changes.• The target text can be entered in free text, but it should be restricted to SNOMED CT categories only.• The system should be to match as much as possible. And display confidence level in the table.

Action items

- New algorithms based on the distance (e.g. edit distance).
- Add the timestamp of each changed mapping result and the details of each change
- Add new ranking results, so the client can make a better judgment of how to edit the wrong result.
- Running benchmark between the distance-based algorithm and RNN overall

01/05/2023 Client Meeting

Date

2023-5-1

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Goals

- Show the client the current progress of this project.
- Get feedback from client.

Discussion items

Item	Notes
Functional improvement	<ul style="list-style-type: none">• The distance between the mapping result and raw text should also be included in the output file• do not filter too many non-medical words• keep the snomed result
Performance	<ul style="list-style-type: none">• speed is not very important

Action items

-
-

08/05/2023 Client Meeting

Date

2023-5-8

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA
- Dr Mike Conway (client)

Goals

- Show the client the current progress of this project again for another client.
- Show the debug result to client.
- Discuss the feedback from the client.

Discussion items

Item	Notes
Functional improvement	Added the ability to change the deployment location of the docker image container

Action items

- Added the function to change the deployment location of the docker image container

15/05/2023 Client Meeting

Date

2023-5-15

Attendees

- Dr Vlada Rozova (client)
- Dr Mike Conway (client)
- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Partial process explanation of mapping algorithm
- About Sprint 3 Planning and Scheduling

Discussion items

Item	Notes
Software Installation	<ul style="list-style-type: none">• Software installation is too cumbersome. Need to simplify the installation process.
Algorithm Process Introduction	<ul style="list-style-type: none">• The algorithm flow is too complicated, and each step needs to be explained in detail, and the corresponding example should be constructed• Client wants to have a set of simple and clear diagrams to show our mapping algorithm.
Function	<ul style="list-style-type: none">• To move the docker image, the container needs to be turned into a button, and the user cannot enter the command line to complete it.• The system needs adopt the latest UIL if the UIL changes in the future

Action items

- To move the docker image, the container needs to be turned into a button, and the user cannot enter the command line to complete it.
- Installation needs to be simpler, click install, and don't make the user type a command to install.
- Need a button or window for users to upload new UIL and download past UIL
- The algorithm flow is too complicated, and each step needs to be explained in detail and the corresponding example should be constructed

22/05/2023 Client Meeting

Date

2023-5-22

Attendees

- Dr Vlada Rozova (client)
- Dr Mike Conway (client)
- Mauro Mello Jr
- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Partial process explanation of mapping algorithm
- About this week's presentation

Discussion items

Item	Notes
Software Installation	<ul style="list-style-type: none">• Demonstrates the simplified installation process.• The installation location of the docker image cannot be changed.
Algorithm Process	<ul style="list-style-type: none">• Showed a set of simple and clear diagrams to demonstrate our mapping algorithm to clients.

Action items



Supervisor Meetings

- [09/03/2023 Meeting notes](#)
- [17/03/2023 Meeting notes](#)
- [30/03/2023 Meeting notes](#)
- [07/04/2023 Meeting notes](#)
- [20/04/2023 Meeting notes](#)
- [27/04/2023 Meeting notes](#)
- [12/05/2023 Meeting notes](#)
- [18/05/2023 Meeting notes](#)
- [25/05/2023 Meeting notes](#)

09/03/2023 Meeting notes

Date

2023-3-9

Attendees

- Mauro Mello Jr
- Yanxi KE
- Jiacheng WANG
- Runzhe HUA
- Mingjun GAO
- Lang Cao

Goals

- The first meeting with the supervisor. He introduced us to this project and the clients.

Discussion items

Item	Notes
Self-introduction of Mauro Mello Jr	His previous experience and current work
Aim of the project	Developing Knowledge, skills, and expertise
Some technical points	The PDCA Cycle, Testing, Learning, Professionalism matters, etc.
About clients and the project	<ul style="list-style-type: none">• Information about clients and Email of them are provided• Mentioned the template has to be used to contact clients• All the requirements will come from clients• Must have first meeting with clients and then ask future questions

Action items

- Contact the customer as soon as possible for more information about the project.

17/03/2023 Meeting notes

Date

2023-3-17

Attendees

- @Mauro Mello Jr
- @Yanxi KE
- @Jiacheng WANG
- @Runzhe HUA
- @Mingjun GAO
- @Lang Cao

Goals

- Communicate with the supervisor about our current progress and receive some advice from him.

Discussion items

Item	Notes
External dependency	<ul style="list-style-type: none">• We still not get the license.• The Universal Indication List is still not to get.• Other alternatives to consider while waiting for these.
Confluence	<ul style="list-style-type: none">• The supervisor emphasized the importance of documenting all documents in Confluence.• Done what can be done currently.• Setting a deadline for the license
Communication	<ul style="list-style-type: none">• Better to have in-person conversations• Planning has the next meeting in person.• Asking may have other solutions

Action items

- Inform stakeholders of the importance of obtaining the license and the full list.
- Schedule another meeting with the client

30/03/2023 Meeting notes

Date

2023-3-30

Attendees

- @Mauro Mello Jr
- @Yanxi KE
- @Jiacheng WANG
- @Runzhe HUA
- @Mingjun GAO
- @Lang Cao

Goals

- Report on the team's latest progress
- Discuss the problems encountered
- Obtain feedback from the supervisor

Discussion items

Item	Notes
Prototype display	<ul style="list-style-type: none">• Report back to the supervisor on the client's view of the prototype• The supervisor believes the UI should conform to the University of Melbourne's design language (if required)
Technical issues	<ul style="list-style-type: none">• The supervisor can assist in a range of medical matters.• For the technical aspects, we should solve them ourselves or ask the technical director.
Future planning	<ul style="list-style-type: none">• The next Meeting time confirmed• Supervisors have the opportunity to have a discussion face-to-face• Discussing the future development process and delivery of projects and reporting on the current development process

07/04/2023 Meeting notes

Date

2023-4-7

Attendees

- @Mauro Mello Jr
- @Yanxi KE
- @Jiacheng WANG
- @Runzhe HUA
- @Mingjun GAO
- @Lang Cao

Goals

- Report on the team's latest progress
- Discuss the problems encountered
- Obtain feedback from the supervisor

Discussion items

Item	Notes
Things to watch out	<ul style="list-style-type: none">• The importance of evidence-based assessments and standardization in document review• Communication issues within project teams regarding project understanding• User acceptance testing as a means to confirm correctness.
Report our progress	<ul style="list-style-type: none">• Describes the main steps and activities of the project until now
Some issues	<ul style="list-style-type: none">• There is not or we do not find a dictionary or library, which only contains medical words.

20/04/2023 Meeting notes

Date

2023-4-20

Attendees

- @Mauro Mello Jr
- @Yanxi KE
- @Jiacheng WANG
- @Runzhe HUA
- @Mingjun GAO
- @Lang Cao

Goals

- Discuss the progress of the project and address various issues

Discussion items

Item	Notes
the progress of the project	<ul style="list-style-type: none">• supervisor asks for a project status update, whether the team is on time and developing things properly.• The client's requirements are being met• A new functionality has been suggested by the client
	we need to export and upload the sprint2 files.
Reviewing Feedback	It is important to provide logical reasoning and tangible evidence to support their disagreements with the marker. need to justify their position rather than simply disagreeing with the mark given.

Action items



27/04/2023 Meeting notes

Date

2023-4-27

Attendees

- @Mauro Mello Jr
- @Yanxi KE
- @Jiacheng WANG
- @Runzhe HUA
- @Mingjun GAO
- @Lang Cao

Goals

- Report to the supervisor about our week's progress

Discussion items

Item	Notes
progress of this week	<ul style="list-style-type: none">• We report the progress of our project and documentation on the confluence.
supervisor's suggestion	<ul style="list-style-type: none">• Confluence documents only export the final version of the content when submitted.

Action items

- Organize documentation according to the supervisor's suggestions

12/05/2023 Meeting notes

Date

2023-4-27

Attendees

- @Yanxi KE
- @Lang Cao

Goals

- Demo products from our group

Discussion items

Item	Notes
discuss	

18/05/2023 Meeting notes

Date

2023-5-18

Attendees

- @Lang Cao

Goals

- Demo products from our group
- Discuss communication issues with customer service with the supervisor.

Discussion items

Item	Notes
Discuss next week's presentation	project introduction.

25/05/2023 Meeting notes

Date

2023-5-25

Attendees

- @Mauro Mello Jr
- @Yanxi KE
- @Jiacheng WANG
- @Runzhe HUA
- @Mingjun GAO
- @Lang Cao

Goals

- About the final presentation

Discussion items

Item	Notes
Show our presentation to the supervisor	
The supervisor proposes amendments.	

Teams Meetings

- 13/03/2023 Team Meeting
- 15/03/2023 Team Meeting
- 17/03/2023 Team Meeting
- 22/03/2023 Team meeting
- 26/03/2023 Team meeting
- 29/03/2023 Team Meeting
- 19/04/2023 Team Meeting
- 26/04/2023 Team Meeting
- 25/05/2023 Team Meeting
- 07/06/2023 Team Meeting

13/03/2023 Team Meeting

Date

2023-3-13

Attendees

- Mingjun GAO
- Lang Cao
- Yanxi KE
- Runzhe HUA
- Jiacheng WANG

Goals

- To discuss the roles in our team.
- To allocate the tasks.

Discussion items

Time	Item
30 mins	<ul style="list-style-type: none">• Introduction and know each other• Collect and understand the requirement of the client• Share the understanding of the project• Know the coding ability of each other• Have an overall understanding of the project• Collect and share the information we have

Question to Clients

- Can we have a sample for raw plaintext
- An account with the license will be provided or not.
- Mapping accuracy requires or not.

Action items

- Research ontoserver API
- Contacting customers for more information and support

15/03/2023 Team Meeting

Date

2023-3-15

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Discussion items

User story

- Collect user requirements together
- Analysis client requirement
- Divide the requirement into three main parts
- Making the initial user story by these parts
- Assign an equal task to each user story
- Record the user story by the user story template

Mapping system

- The mapping system has two methods based on the current situation
- Ontoserver license application started.

Action items

- Write effective user stories.
- Appy Ontoserver account

17/03/2023 Team Meeting

Date

2023-3-17

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Goals

- Regular progress updates and planning

Discussion items

Item	Notes
Discussed the Sprint 1 Assessment Checklist.	<ul style="list-style-type: none">• Discuss the difficult and important parts of the sprint one checklist• Confirm the requirement on the current stage• Share the understanding of software development• Assign tasks to individuals.
The website design prototype	<ul style="list-style-type: none">• Discuss how many pages are needed on this website• Discuss the function on each page• Discussed the design for the front-end pages needed for the current stage.
Mapping System	<ul style="list-style-type: none">• Updating the current situation on the mapping system<ul style="list-style-type: none">• Ontoserver license still is not available in the current stage• the Universal Indication List still be prepared.
Main decision	<ul style="list-style-type: none">• Decide to do the website first

Action items

- Runzhe HUA is responsible for organizing the meeting minutes and the design of the website's index.
- Lang Cao needs to write the background description, client goals, and motivation sections. At the same time, he is also responsible for designing the pages for normal users.
- Jiacheng WANG needs to write personas and design the interface for moderator information.
- Yanxi KE is responsible for to analysis of requirements and design admin page.
- Mingjun GAO is responsible for developing a plan for Sprints 2 and 3, including requirements, technologies, and infrastructure, with estimated and prioritized backlog items available.

22/03/2023 Team meeting

Date

2023-3-22

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Goals

- Regular progress updates and planning

Discussion items

Item	Notes
Go through the checklist for spring 1	<ul style="list-style-type: none">• Discuss Each Question on the checklist• Check if there is missing or not
The website design prototype	<ul style="list-style-type: none">• Discuss how to the implementation of the webpage
Prepare Questions for client meeting	<ul style="list-style-type: none">• Prepare Question so that need to discuss in the meeting with Clients

Prepare Question

1. OntoServer still cannot use
 2. UIL is empty
 3. There are two parts to this project
1. Account system
 - a. doctors, researchers, and admin
 - a. Modify mapping result?
 - a. Login, register, and change information(name, phone number),
b. who will be the users?
 - c. Does the admin need to update UIL?
 - d. Does the admin need to see each mapping?
 - e. if there are multiple mapping systems, should the admin can back up the weight as a setting checkpoint?
 2. Mapping system
 - a. confirmation, unsure, cannot
 - a. Based on the Score
 - b. Could you provide the list already be mapped
 3. Do not use Ontoserver and build the web in docker.
 4. Discuss meeting time.

26/03/2023 Team meeting

Date

2023-3-26

Attendees

- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO

Goals

- Discussed the simple matching algorithm that has been implemented.
- Assigned tasks for the next stage.

Discussion items

Item	Notes
Matching algorithm	<ul style="list-style-type: none">• Lang Cao presented the current matching algorithm to the team members.• Discussed the algorithm and agreed that using ontoserver could improve the matching accuracy.• Ontoserver is currently still unavailable and no other replacements have been found.
Assign tasks	<ul style="list-style-type: none">• There are currently three main pages on the frontend.• Three people in the team are assigned to develop frontend pages respectively.• The rest of the team will continue to optimize the algorithm.

Action items

- Yanxi KE needs to build a page for upload and download.
- Runzhe HUA needs to visualize the results.
- Jiacheng WANG needs to develop a page showing the mapping results list.

29/03/2023 Team Meeting

Date

2023-3-29

Attendees

- Runzhe HUA
- Lang Cao
- Yanxi KE
- Jiacheng WANG

Goals

- Discussed and tested the pages we've made

Discussion items

Item	Notes
Web pages	<ul style="list-style-type: none">• We show the three web pages we made before.• Tested basic functions of each page.• The function of modifying the mapping result has not been perfected yet, so we discussed this.
Mapping result	<ul style="list-style-type: none">• Need to improve the accuracy of the algorithm but no ground truth• Decided to try manual mapping

Action items

- merge branches together
- try to map manually

19/04/2023 Team Meeting

Date

2023-4-19

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Goals

- Discussed and installed the environment of our project.
- Future needs

Discussion items

Item	Notes
Environment	<ul style="list-style-type: none">• We install the environment for each teammate that the algorithm needs
Medical Dataset	<ul style="list-style-type: none">• We need to find out more useful medical datasets for algorithm
JSON structure	<ul style="list-style-type: none">• We need to transform the data to a JSON structure. because it's more useful for data transmission and display during service. This format is more universal and convenient.
Algorithm	<ul style="list-style-type: none">• There may be bottlenecks in the algorithm at present, which may require the accumulation of databases and research on other more suitable algorithms to achieve significant improvement. Therefore, there will not be too much R&D for the time being.
Confluence	<ul style="list-style-type: none">• Currently, the documents on Confluence are still too thin, and it is necessary to continue enriching the documentation.

Action items

- Implement a method for converting data formats to JSON format and outputting a JSON file to the server in the "Writing.py" file.
- Find out more useful medical database
- do more confluence document

26/04/2023 Team Meeting

Date

2023-4-26

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Goals

- Discussed and installed the environment of our project.
- Sprint 3 Planning
- Sprint 2 Review

Discussion items

Item	Notes
Sprint 2 Review	<ul style="list-style-type: none">• Discuss the review. details in this link. Sprint2 - Review
Sprint 3 Plan	<ul style="list-style-type: none">• Discuss what going to do. Details are in this link. Sprint 3 Plan
Future	<ul style="list-style-type: none">• Validation Set Testing• Client Requirements Implementation• Continued Documentation Enhancement

Action items

- Finish sprint 2 review
- Finish sprint 3 plan

25/05/2023 Team Meeting

Date

2023-5-25

Attendees

- Lang Cao
- Yanxi KE
- Jiacheng WANG
- Mingjun GAO
- Runzhe HUA

Goals

- Discussed and design the Sprint 4 plan
- Discussed the presentation
- Discussed the checklist of sprint 4
- Review the Sprint 3

Discussion items

Item	Notes
Sprint 3 review	<ul style="list-style-type: none">• The work completed, the challenges faced• Identifying areas of improvement and learning for the next Sprint.
Sprint 4 plan	<ul style="list-style-type: none">• Setting objectives, assigning tasks, and establishing timelines.• Packaging and releasing a stable version in Sprint4• Code optimization, machine learning performance enhancement
Presentation	<ul style="list-style-type: none">• Discuss the forthcoming presentation, deliberating on the content, format• Ensure a concise and engaging representation of the project's progress

Action items

- Finish sprint 3 review
- Finish sprint 4 plan

07/06/2023 Team Meeting

Date

2023-6-7

Attendees

- Lang Cao
- Mingjun GAO
- Yanxi KE
- Jiacheng WANG
- Runzhe HUA

Goals

- Review Sprint 4 final product and document
- Review Feedback from Clients

Discussion items

Item	Notes
Sprint 4 review	<ul style="list-style-type: none">• Packaging and releasing a stable version of the project.• Optimizing the codebase for better readability and maintainability, and preparing accompanying documentation.• Enhancing the performance of existing machine learning algorithms.• Refining the user manual for better user understanding and engagement.
Feedback Discussion	<ul style="list-style-type: none">• Overall project satisfaction and customers' ideas for future development.• A suggestion for conducting real-world deployment tests post server permissions.• The request for more comprehensive documentation, and acknowledgement of the current satisfactory status of the codebase.

Action items

- Finish sprint 4 review
- Necessary documentation improvements
- Final touches based on the feedback received

Development

This page houses documentation related to the development aspect of Digital Health.

- [Developing Milestones](#)
- [Development Environment](#)
- [Deployment Plan](#)
- [GitHub Version Control](#)
- [Guide](#)
- [Technical Analysis](#)
- [GitHub Version Control](#)
- [Mapping System](#)

Guide

Development environment requirements (keep updating)

- Python 3.5 or above
 - Flask
 - Panda
 - Numpy
 - NLTK
 - sci-spacy
 - openpyxl
 - Flask-cors
 - torch
 - pymysql
 - torch
 - transformer
 - autoModel
 - editdistance
 - mpi4py
 - JSON
- Node.js
 - serve
 - element-ui
 - echarts
 - axios

You could use pip install [package_name], to install the Python package you don't have.

For sci-spacy, the model may need to be installed first. How to download and install a sci-spacy pre-trained model can be viewed here: [scispacy | SpaCy models for biomedical text processing \(allenai.github.io\)](#). We currently use the model: en_ner_bionlp13cg_md.

The file that could be processed:

- .txt
- .csv

Note: The file you submitted should be a single-column note describing the medical note. The result is in JSON.

How to start the server:

Running the Docker first (details and picture in mapping algorithm page)

Change your current to ~/DI-RedBack/src and run the command "server -s dist"(may require administrator privileges) and

"python main.py", then a local server will be created on the following URL: "<http://127.0.0.1:3000>". And all the features will be on this website.

Development Environment

Front-end

Node

Download and install nodejs from <https://nodejs.org/en>

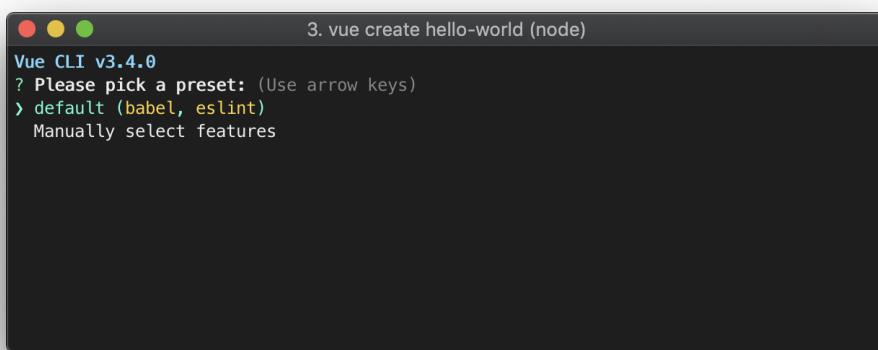
Install vue-cli

```
npm install -g @vue/cli
```

Creating a Vue Project

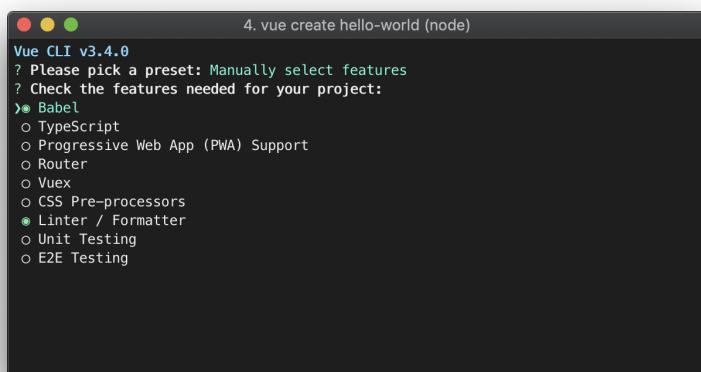
Input vue create project-name in shell

Choose manually select features, Vue version 2.X



```
3. vue create hello-world (node)
Vue CLI v3.4.0
? Please pick a preset: (Use arrow keys)
❯ default (babel, eslint)
Manually select features
```

Select Babel and Router



```
4. vue create hello-world (node)
Vue CLI v3.4.0
? Please pick a preset: Manually select features
? Check the features needed for your project:
❯ Babel
  o TypeScript
  o Progressive Web App (PWA) Support
  o Router
  o Vuex
  o CSS Pre-processors
  ● Linter / Formatter
  o Unit Testing
  o E2E Testing
```

Install packages using npm

```
npm install echarts axios element-ui
```

Git clone from branch sprint2-milestone2 and changes the current working directory to src/frontend

Replace vue initial files with files in src/frontend

Run front-end server

```
npm run server
```

Back-end

Install python=3.7

Python package requirements

```
pip install transformers bcrypt flask flask-cors pymysql spacy scipy torch mpi4py requests pandas cryptography https://s3-us-west-2.amazonaws.com/ai2-s2-scispacy/releases/v0.5.1/en\_ner\_bionlp13cg\_md-0.5.1.tar.gz
```

git clone branch sprint2-milestone2 and changes the current working directory to src/backend

Running back-end server

```
flask --app main run
```

Database

Download docker and git clone sprint2-milestone2 and changes the current working directory to the SRC/DOCKER-SETUP/DOCKER-MYSQL path

Input "docker-compose up -d"

Connecting MySQL in Flask

```
con = pymysql.connect(  
    host='mysql',  
    port=3306,  
    user='root',  
    password='123456',  
    db='mapping_system',  
)
```

Ontoserver

Log in to [quay.io](#) with:

```
docker login quay.io
```

Minimal Configuration

- Create a text file called docker-compose.yml with the following contents (or download [./docker-compose.yml](#)).
- If deploying in Australia, details on configuring NCTS_CLIENT_ID and NCTS_CLIENT_SECRET can be found [here](#).

Otherwise, you only need to include this configuration if you have a secured upstream syndication service. In this case, the process for obtaining credentials will be specific to your syndication provider.

- You will almost certainly want to set the JVM maximum heap via JAVA_OPTS=-Xmx as per below, as the defaults can be unreliable.
- NOTE: By default, Ontoserver will run using SSL/TLS (i.e. https://). To disable SSL/TLS, add - ONTO SERVER_INSECURE=true to the environment section of the ontoserver container in the docker-compose.yml file.

```

version: '3'
volumes:
  onto:
    driver: local
  pgdata:
    driver: local
services:
  db:
    image: postgres:12
    volumes:
      - pgdata:/var/lib/postgresql/data
  healthcheck:
    test: ["CMD-SHELL", "pg_isready -U postgres"]
    interval: 10s
    timeout: 5s
    retries: 5
  environment:
    - POSTGRES_HOST_AUTH_METHOD=trust
ontoserver:
  image: quay.io/aehrc/ontoserver:ctsa-6
  container_name: ontoserver
  read_only: true
  security_opt:
    - no-new-privileges
  depends_on:
    - db
  ports:
    - "8443:8443"
    - "8080:8080"
  environment:
    - spring.datasource.url=jdbc:postgresql://db/postgres
# These two lines are specific to deployment in Australia only
  - authentication.oauth.endpoint.client_id.0=NCTS_CLIENT_ID
  - authentication.oauth.endpoint.client_secret.0=NCTS_CLIENT_SECRET
#      - JAVA_OPTS=-Xmx2G          # Minimum
#      - JAVA_OPTS=-Xmx8G          # Preferred
  volumes:
    - onto:/var/onto
    - /tmp
    - /var/log

```

Running Ontoserver

- Starting

docker-compose up -d

- Stopping

docker-compose stop

- Download the latest SNOMED CT-AU version

docker exec ontoserver /index.sh

- Download a specific SNOMED CT-AU version

docker exec ontoserver /index.sh -v 20230228

Deployment Plan

Document Status	Version	Date	Comment
DONE	3.0	2023-5-23	Docker deployment.
DONE	2.0	2023-4-24	Adding local deployment
DONE	1.0	2023-3-19	Adding Ontoserver deployment.

System Requirements

1. 8GB memory
2. At least 10GB of storage space is required
3. Need Internet for the first time install everything.

Download and Install Docker on Windows

<https://docs.docker.com/desktop/install/windows-install/>

Code clone

```
git clone branch sprint3
```

Deployment

```
cd src
```

```
docker login quay.io
```

Running docker compose

- Starting

```
docker-compose up -d
```

Docker Desktop Upgrade plan Search for local and remote images, containers, and more... Ctrl+K Sign in

Containers Give feedback

A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. [Learn more](#)

Only show running containers Search

	Name	Image	Status	Port(s)	Last started	Actions
src	-	Exited	1 minute ago			
ontoserver	quay.io/aehrc/ontoserver:cts	Exited (255)	2 days ago	8080:8080	Show all ports (2)	
db-1	postgres:12	Exited (255)	2 days ago			
mysql-1	src-mysql	Exited	1 minute ago	3306:3306		
node-1	src-node	Exited (255)	1 day ago	3000:3000		
flask-1	src-flask	Exited (255)	1 day ago	5000:5000		

Showing 6 items

RAM 1.66 GB CPU 0.03% Not connected to Hub v4.17.0

- Download the latest SNOMED CT-AU version

docker exec ontoserver /index.sh

Go to this link [http://localhost:8080/fhir/ValueSet/\\$expand?url=http://snomed.info/sct?fhir_vs=refset/32570071000036102&count=10&filter=cough](http://localhost:8080/fhir/ValueSet/$expand?url=http://snomed.info/sct?fhir_vs=refset/32570071000036102&count=10&filter=cough).

If the expansion content is same as below, it means snomed-ct has been downloaded successfully:

Expansion

Size: 82

version	http://snomed.info/sct http://snomed.info/sct/32506021000036107/version/20230430
displayLanguage	en-US,en;q=0.9,zh-CN;q=0.8,zh;q=0.7
count	10
offset	0
filter	cough

Code	Display	System	Inactive	Version
49727002	Cough	http://snomed.info/sct	false	
300959008	Allergic cough	http://snomed.info/sct	false	
17986004	Barking cough	http://snomed.info/sct	false	
62427007	Bovine cough	http://snomed.info/sct	false	
68154008	Chronic cough	http://snomed.info/sct	false	
289965001	Croupy cough	http://snomed.info/sct	false	
289114008	Does cough	http://snomed.info/sct	false	
11833005	Dry cough	http://snomed.info/sct	false	
59994004	Hacking cough	http://snomed.info/sct	false	
43025008	Paroxysmal cough	http://snomed.info/sct	false	

otherwise run "docker exec ontoserver /index.sh" again.

Developing Milestones

Sprint 2 (24 Mar ~ 19 May)

Date	GitHub Branch	What major features we have done	What next	Clients feedback
16/04/2023	sprint2_milestone2	<ul style="list-style-type: none"> Greater integration of functions Successfully deployed and integrated the backend (Flask) with MySQL Better visualization and data modification effects and functions (delete, search) Preliminary completion of a relatively slow RNN-based algorithm (transformer) Added registration and login, and accompanying user information features History mapping also can delete and modify 	<ul style="list-style-type: none"> Adding status for processing so that can upload multiple files Change transformer model Improve the mapping performance 	<ul style="list-style-type: none"> This time the client is very satisfied with the current interface and matching principles Need to add everyone's match history instead of the current user's Need to add a modified history Use edit distance to show the performance of matching results
31/03/2023	sprint2_milestone1	<ul style="list-style-type: none"> Enabled users to modify and save mapping results Allowed users to upload raw data and download mapping results Achieved fairly accurate mapping results Implemented basic data visualization The overall project can already be fully built on a local server rather than a scattered set of finished functions 	<ul style="list-style-type: none"> Improve the efficiency and accuracy of algorithms Improve the consistency UI Obtain larger dataset Testing for the main functions Modifications based on customer feedback Building DataBase for storage of the mapping history and login 	<ul style="list-style-type: none"> Overall, the client is very satisfied with the current results and progress Need to modify a suitable interface Need to be able to view the history of mapping Introduce distance to judge matching performance

Milestone 2

Challenge

1. Few Transformer model suit medical data
2. Having some mapping requirements unclear



Login

User Name

Password

Remember Me

Login

Don't have an account? [Sign Up](#)



Register

* Username

* Email

* Password

* Confirm Password

Register

Mapping System **Upload** Mappings Hi, admin1

txt file with a size less than 200Mb

select file **upload to server**

Mapping System **Upload** Mappings Hi, admin1

Process

Mapping System Upload Mappings

ID	Title	Description	Created At	Type to search
f660b2a4-8469-4544-a3ae-ff6167f0cc7f			2023-04-16 22:55:58	View Download Delete

Hi, admin1

Mapping System Upload Mappings

Mapping Result

Mapping Source

id	Raw Text	Target Text	Source	Type to search
0	Sinusitis	Sinusitis	SNOMED CT	Edit Delete
1	Tonsillitis	tonsillitis	SNOMED CT	Edit Delete
2	Skin infection	Infection of skin	SNOMED CT	Edit Delete
3	LRTI	Lower respiratory tract infection	SNOMED CT	Edit Delete
4	Bacterial vaginosis	bacterial vaginosis	SNOMED CT	Edit Delete
5	Review	Non-Match	URL	Edit Delete
6	??UTI	Lower urinary tract infectious disease	SNOMED CT	Edit Delete

Working description

1. We have a new login and register page with the University of Melbourne icon.
2. All of the data are stored in the database.
3. New visualisation and view mapping result page
 - a. users can see their history and modify their history mapping data.
 - b. They can search the specific result
 - c. there is a filter so that can find a kind of result.
4. Update the new mapping algorithm (Version 1 on the mapping algorithm page.)

Milestone 1

Challenge

1. Searching on SNOMED-CT is very slow, 0.6 seconds for one search, but not using all the computer resources in the process.
2. This is the first time that we connected to MySQL by Flask.
3. In natural language, few libraries or models are optimized for medical symptoms. This leads to the misrecognized meaning or the unrecognized null value returned.
4. Many diseases are founded in SNOMED CT but not mentioned in UIL at all.

Working Details

DI TEST

Upload

Drag or Click to upload Files

Process

Upload

Drag or Click to upload Files

ReasonExample.txt

Process

This is the load function. The user can select the raw text here and then click the mapping then the system starts mapping the raw text to the UI.

Raw	Result
Sinusitis	Non-Match
Tonsillitis	Tonsillitis
Skin infection	Skin and soft tissue infection, other
LRTI	Non-Match
Bacterial vaginosis	Bacterial vaginosis
Review	Non-Match
UTI/UTI	Non-Match
Dental prophylaxis	Non-Match
Infective exacerbation of asthma	Asthma, infective exacerbation
Atypical pneumonia	Non-Match
Travel advice	Non-Match
"Sinusitis, frontal"	Non-Match
Personal dermatitis	Personal dermatitis
Administrative procedure	Surgical prophylaxis, gynaecological procedure
Prescription	Non-Match
Personal abscess	Personal / anorectal abscess
Probable Hypertension - Borderline	Non-Match
Travel vaccination	Non-Match
"Cough, post infective"	Non-Match
STI screen	Non-Match
Vaginal discharge - episiotomy	Non-Match
"Cellulitis, leg"	Non-Match
Duodenitis	Duodenitis
URTI	Non-Match
Left Eye injury	Penetrating eye injury
"URT, w/o"	Non-Match
Left Orbit media	Non-Match
Keratoacanthoma	Non-Match

This is the mapping result. The left side is the raw text and the right side is the mapping result

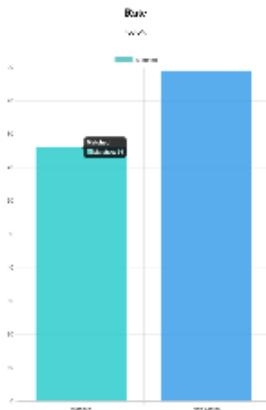
Raw	Result
Sinusitis	Subusitis
Tonsillitis	Tonsillitis

127.0.0.1:5000
Referencing Roadm... GitHub Setting up your De... 提交计划申请 - UT... Dashboard The University 127.0.0.1:5000 显示
Successfully updated modify.csv 确定

ReasonExample.txt

Process	Download	Visualization	Raw	Result
Sinusitis			Sinusitis	Subusitis
Tonsillitis			Tonsillitis	Tonsillitis
Skin infection				Skin and soft tissue infection, other
LRTI				Non-Match
Bacterial vaginosis				Bacterial vaginosis
Review				Non-Match
??UTI				Non-Match
Dental prophylaxis				Non-Match

The interface allows the user to modify the mapping result. If the result has been mapped, it will change colour to red. If click confirm, it will change to green.



This is for the visualisation of the number of Matched and Non-Match results.

Sprint 3

Technical Analysis

This Confluence page presents a comprehensive overview of the various technology solutions chosen for a medical text analysis project. It covers algorithm selection, front-end and back-end frameworks, databases, virtual machines, and other related components. The Mapping Algorithm section has been given particular attention and expanded upon on a separate page, as it plays a critical role in the success of the medical text analysis project.

Mapping Algorithm

Requirements	Algorithms options	Description
Suitable for Data analyze	PCA, k-Means, t-SNE	Commonly used for data analysis, allowing for dimensionality reduction, clustering, and visualization.
Easy and light with	Linear Regression, Naive Bayes	Simple and easy-to-understand algorithms with relatively low computational cost.
Good scalability and performance	Random Forest, XGBoost	Ensemble methods provide high scalability and performance, suitable for large datasets and complex tasks.
Been tested on various medical tasks	SVM, LSTM, Convolutional Neural Network	Widely applied in medical tasks, such as medical image classification, time-series analysis, and NLP.

Front-End

Requirements	Front-End Framework Options	Description
Simple and direct	Vue.js, React, Angular	Our team has experience using Vue.js, making it the preferred option. React and Angular are also popular and widely-used options, but our team is less experienced with these frameworks.
User-friendly	Vue.js, React, Angular	Vue.js has a wide range of documentation and plugins available, making it easy to use and learn. React and Angular are also user-friendly, but Vue.js was selected based on our team's experience.
Display system efficiency and accuracy using charts	Vue.js, React	Both Vue.js and React have a variety of charting libraries available, making it easy to display system efficiency and accuracy using charts. Angular is also an option but was not chosen due to our team's limited experience.
Lightweight language	Vue.js, React	Both Vue.js and React are lightweight front-end frameworks, making them well-suited for web development. Angular is also an option but can be more complex and heavyweight.
Easy to use and reliable	Vue.js, React, Angular	Vue.js and React are both known for being easy to use and reliable. Angular can also be reliable but may require more expertise and experience to implement effectively.

Back-End

• Back-End Framework

Requirements	Back-End Framework Options	Description
Easy to Learn	Flask, Pyramid, Bottle	Flask is simpler and easier to learn compared to Pyramid and Bottle. Flask also has a larger community and more comprehensive documentation, making it easier for new developers to get started.
Lightweight for Small Projects	Flask, Bottle	Flask and Bottle are both lightweight frameworks that are more suitable for small projects compared to more comprehensive options like Django and Pyramid. Flask and Bottle allow for more flexibility in development and can be easier to set up and configure.
Flexibility	Flask, Pyramid, Falcon	Flask, Pyramid, and Falcon are all known for their flexibility in development. Flask and Pyramid allow for more customization in terms of configuration and middleware, while Falcon is known for its speed and ability to handle high-performance applications.
Availability of Extensions and	Flask, Pyramid,	Flask, Pyramid, and Django are all popular frameworks with a wide range of extensions and middleware available. Flask and Pyramid have a more comprehensive range of options available, while Django has a larger community.

Middleware

Django

and a wide range of third-party integrations available.

• Database

Requirements	Options	Description
Open-source and free	MySQL, PostgreSQL, MongoDB	MySQL is an open-source and free database framework that meets the specified requirements. PostgreSQL and MongoDB are also open-source and free but were not chosen due to our team's greater experience with MySQL.
Cross-platform compatibility	MySQL, PostgreSQL	Both MySQL and PostgreSQL are cross-platform compatible, making them suitable for deployment on a variety of platforms.
High performance and reliability	MySQL, PostgreSQL	Both MySQL and PostgreSQL are known for their high performance and reliability, making them suitable for handling large amounts of data and high-traffic applications.

Virtual Machine

Requirements	Options	Description
Containerization Technology	Docker	Docker is the preferred containerization technology requested by the client for deployment due to its ease of use, portability, and scalability. Additionally, Docker's containerization technology can help to isolate applications and their dependencies, which can reduce the risk of conflicts and improve system stability.
Virtual Machine	VirtualBox, VMware, Hyper-V	Virtual machines such as VirtualBox, VMware, and Hyper-V can provide a stable and isolated environment for running the application on a local machine.
Cloud Service Provider	AWS, Google Cloud, Azure	Cloud service providers such as AWS, Google Cloud, and Azure can provide scalable and flexible deployment options.
Operating System	Linux, Windows, MacOS	The operating system for deployment can vary depending on the specific needs and requirements of the project. Linux is a common choice for server deployment, while Windows and MacOS can also be used depending on the application's compatibility and requirements.

GitHub Version Control

Development Branch

Each team member completes his own part under his own branch and then uses the merge method. The naming rules are sprint number and your own name. In sprint 2, there are the following

- sprint2-Jiacheng
- sprint2-Runzhe2
- sprint2_yanxi
- sprint2_lang
- sprint2_gfy
- sprint3_gfy
- sprint3_lang

Milestone Branch

There are three milestones included. The branch named after Milestone is a version that implements as many functions as possible under the current situation and has significant improvements.

In sprint 2 are the following

- sprint2_milestone1
- sprint2_milestone2

In sprint 3 is the following

- sprint3_milestone3

Final Branch (For testing or delivery)

- sprint3_alpha1
- sprint3_beta1
- sprint4_beta1

Mapping System

- Mapping algorithm
- Mapping analysis
- Mapping Scenario

Mapping algorithm

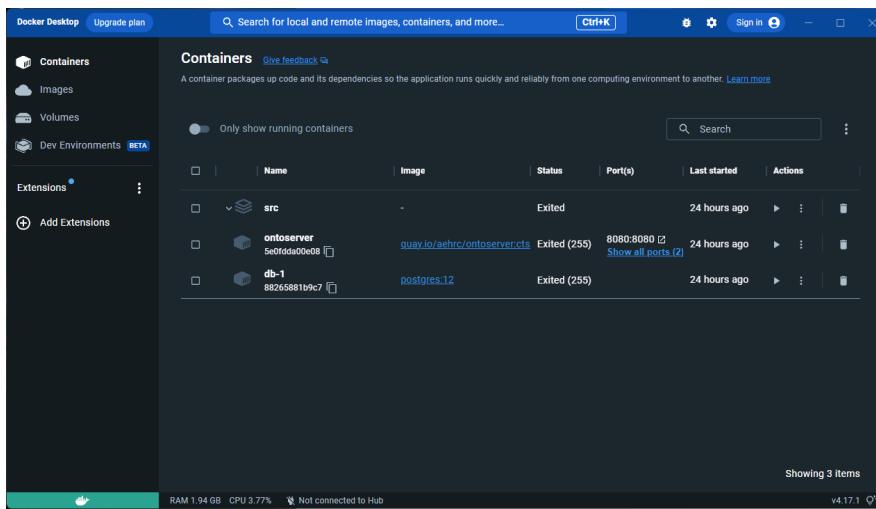
There are several technologies used here. Each of them will be introduced and analysed. The discussion section will introduce how these technologies are used in this project. Then the reason that using this technology will be explained and the advantages and disadvantages will be discussed.

OntoServer-SNOMED CT and Docker

OntoServer is used for accessing the SNOMED CT database through the [quay.io](#) account. Docker Desktop is the software platform used for running the server.

SNOMED CT is a systematically organized and computer-friendly medical terminology collection covering most aspects of clinical information, such as diseases, findings, operations, microorganisms, drugs, etc. The terminology enables consistent indexing, storage, retrieval, and aggregation of clinical data across disciplines, specialties, and sites of care. In the current project, the SNOMED CT vocabulary specific to medical conditions in Australia will be used.

Docker is an open-source project implemented in the Go language, which allows us to easily create and use containers, docker packages the program and all the dependencies of the program into the docker container so that your program can have consistent performance in any environment, where the dependence of the program to run is the container is like a container, the operating system environment where the container is located is like a cargo ship or port, and the performance of the program is only related to the container (container), It has nothing to do with which cargo ship or port (operating system) the container is placed on.



The above screenshot shows the SNOMED CT connected by docker as a local environment. Running the container called src so that it is activated. Use a request link with the information want to search on SNOMED CT, after which a JSON response including all possible matches can be obtained.

Approximately 80% of the text in the example can be searched and returned as the best match by SNOMED CT. However, most of them are not listed in the Universal Indication List, and half of them are only mentioned as similar indications with some relationship to the Universal Indication List.

In addition, searching on SNOMED CT is very slow because it is a huge database. The situation is the following MPI4PY, which will be discussed.

The SNOMED CT database we utilized is available on Australian Medicines Terminology (AMT), which is available to registered users from NCTS (National Clinical Terminology Service) website. To access the database, users need a [quay.io](#) account with a full license, which is held by the university.

Why use this and what advantages and disadvantages

This is specified by the client unless there is a better way. The advantage includes, it is very convenient to use this, only one HTTP request is required, and the results obtained are very accurate. The disadvantage is that it is very difficult to deploy. First, you need to get an authenticated account, and then deploy and use it by reading a lot of official usage documents, and there are errors in the official usage method. In addition, it is very slow in searching.

Scispacy

Scispacy is a Python package that contains spacy models for processing text in biomedical, scientific or clinical domains.



Interactive Demo

just looking to test out the models on your data? Check out our [demo](#).

Installing

```
pip install scispacy  
pip install -e model URLs
```

Models

Model	Description	Install URL
en_core_scibio	A full spacy pipeline for biomedical data.	Download
en_core_scitred	A full spacy pipeline for biomedical data with a larger vocabulary and 50k word vectors.	Download
en_core_scilbert	A full spacy pipeline for biomedical data with a ~7Bk vocabulary and a Tfm1/scilbert-base in the transformer model.	Download
en_core_scilg	A full spacy pipeline for biomedical data with a larger vocabulary and RoBERTa word vectors.	Download
en_ner_craft_md	A spacy NER model trained on the CRRAFT corpus.	Download
en_ner_wikipe_md	A spacy NER model trained on the NLPEA corpus.	Download
en_ner_bionlp13cg_md	A spacy NER model trained on the BIONLP13CG corpus.	Download
en_ner_bionlp13cg_smf	A spacy NER model trained on the BIONLP13CG corpus.	Download

This project is maintained by [alirezai](#)

Hosted on GitHub Pages — Theme by [unitedstack](#)

There are 8 models in Scispacy and en_ner_bionlp13cg_md will be used in this project. It is including AMINO_ACID, ANATOMICAL_SYSTEM, CANCER, CELL, CELLULAR_COMPONENT, DEVELOPING_ANATOMICAL_STRUCTURE, GENE_OR_GENE_PRODUCT, IMMATERIAL_ANATOMICAL_ENTITY, MULTI_TISSUE_STRUCTURE, ORGAN, ORGANISM, ORGANISM_SUBDIVISION, ORGANISM_SUBSTANCE, PATHOLOGICAL_FORMATION, SIMPLE_CHEMICAL, TISSUE. It is a pre-train model based on a recurrent neural network model with an F1 score of 76.57. Given that SNOMED CT's results may not always match the Universal Indication List, we use this model to determine the disease's type by identifying its parent entity.

Why use this and what advantages and disadvantages

The reason for using this is that he can find the same disease type in UIL by looking up the parent class to decide which disease it is. The advantage is that it is very easy to prepare and use, and the accuracy rate is also very high. The downside is that some non-medical words will also find their parents.

Transformer

The Transformer model is a deep learning model that uses a self-attention mechanism, which can assign different weights according to the importance of each part of the input data. This model is mainly used in natural language processing (NLP) and computer vision (CV).

In this project, we will use the Transformer model to encode raw text into a set of vector representations that capture the semantic meaning of the plain text. And then leverage the vector similarity to identify the most appropriate candidate and evaluate the quality of the SNOMED CT's output. Specifically, we will leverage the Transformer's ability to identify relationships between different parts of the input text and generate highly expressive and informative representations that can be easily used for downstream tasks. The effectiveness of this approach is well-established in the previous literature, and we anticipate achieving highly accurate results using this method.

Why use this and what advantages and disadvantages

The reason for using this is because this is an rnn-based pre-trained model that converts text into vectors. Thus the distance of cosine can be calculated. If the final result is too far away, the result is wrong. The advantage of using this is that his calculations are very accurate because 1.3 billion data are used for pre-training. The downside is that the computation is very slow.

MPI4PY

This is a package in Python. The full name of MPI is Message Passing Interface, which is the message passing interface. it is a standard for parallel programs, including protocols and semantic descriptions. They specify how to use their characteristics in various implementations. There are some specific implementations such as MPICH, OpenMPI, etc., and provide Fortran, C, and C++ corresponding interfaces. The goals of MPI are high performance, large scale, and portability. MPI remains the dominant model for high-performance computing today.

Why use this and what advantages and disadvantages

In this project, we use this to operate the request to search on SNOMED CT. Speed up with a parallel approach. The upside is that we end up reducing the runtime by a factor of 7. The downside is that the original code needs to be significantly modified. And because the running instructions are different from before, it is necessary to separate this part to generate a file and re-read it in the main program.

Edit distance

The edit distance is a quantitative measurement of the degree of difference between two strings (such as English characters). The measurement method is to see how many times processing is required to change one string into another. Edit distance can be used in natural language processing. For

example, spell-checking can judge which one (or several) is a more likely word based on the edit distance between a misspelled word and other correct words. DNA can also be regarded as a string composed of A, C, G, and T, so the edit distance is also used in bioinformatics to judge the similarity of two DNAs. The diff and patch under Unix are examples of using edit distance to compare text edits.

Why use this and what advantages and disadvantages

Clients ask to use it in a specific situation to map and use it to show the performance in the visualization. The advantages of using it are that it is quickly and easy to use. The disadvantages are limited, very inaccurate results for words if written.

Technology not used but considered

Numba

Numba is an open-source JIT compiler that uses LLVM to translate a numerically-focused subset of Python, including many Numpy functions, into fast machine code through the llvmlite bindings. It provides a large number of options for parallelizing Python code on CPUs and GPUs, often requiring only minor code changes.

The reason that it is not used is that it cannot contain an HTTP request.

Traditional Recurrent Neural network

A recurrent Neural Network (RNN) is a type of recurrent neural network that takes sequence data as input, performs recursion in the evolution direction of the sequence, and all nodes (circular units) are connected in a chain. Among them, two-way recurrent neural network and long short-term memory networks are common recurrent neural networks.

Recurrent neural networks have memory, parameter sharing, and Turing completeness, so they have certain advantages in learning the nonlinear characteristics of sequences. Recurrent neural networks have applications in natural language processing, such as speech recognition, language modelling, machine translation, and other fields, and are also used in various time series forecasts. The recurrent neural network constructed by introducing the convolutional neural network can deal with computer vision problems involving sequence input.

In this project, we do not have enough data and we do not have ground truth value, therefore we cannot do the training and testing to build the neural network model. In addition, if the model has too many parameters, it will take a lot of time to load.

Nginx

It is an open-source web server that can also be used as a reverse proxy, load balancer, mail proxy and HTTP cache. In this project, because the mapping system needs lots of resources and many users may upload raw text for mapping at the same time, therefore using Nginx to know the loading of the current balance. If the previous access data processing is not over or the server load is very high, Nginx can cache the current request and send the request after the previous data processing is completed to ensure that the server will not crash.

Mapping analysis

Mapping Version 1(Based on GitHub version called 'sprint2_milestone2')

The link is here: [sprint3_alpha1](#)

```
1 Sinusitis
2 Tonsillitis
3 Skin infection
4 LRTI
5 Bacterial vaginosis
6 Review
7 ???UTI
8 Dental prophylaxis
9 Infective exacerbation of asthma
10 Atypical pneumonia
11 Travel advice
12 "Sinusitis, frontal"
13 Perioral dermatitis
14 Administrative procedure
15 Prescription
16 Perianal abscess
17 Probable Hypertension - Borderline
18 Travel vaccination
19 "Cough, post infective"
20 STI screen
21 Vaginal discharge - episiotomy.
```

The example raw text is given by clients. Each line of raw text can be mapped into the UIL.

The following is the entire matching process. In the raw text, there are a large number of noise characters, including '-', '/' and non-alphabetic symbols. When reading the raw text, removing this interference information and saving it to the dictionary is first necessary.

One of the important reading files is the UIL, which is provided by clients. there are four columns of data. All of the letters need to be lowercase, as is the raw text file. Reading the modified file also be needed in this step.

Split the processed data by ' ', '/' and " as tokenisation.

Using the processed data and searching them through SNOMED CT. In the example case, there are approximately 80% of the can be found the result through this method. however, since SNOMED CT is a very large vocabulary, non-disease or symptom words will be matched to the result, which is not what we want.

If the SNOMED CT result is in the UIL, then this is the result and matching will not continue. if the result is not in UIL, then find the parent of the return value from SNOMED CT. If the parent is in UIL, then return the parent as a result. If it is part of the UIL, then add it to the dictionary for counting as a candidate. Then search the UIL and add it to the dictionary as a candidate using processed text. If the highest counter in the dictionary is large the second one, save as the result, otherwise use SNOMED CT as the result. If the dictionary is empty, return it as Non-Match. The pending result uses a transformer to vector and then calculate the distance. If the distance is less than 0.75, means Non-Match, otherwise, the pending result will be the final result. In addition, the search system not only returns a UIL as a result but also return a list that contains all of the probable mapping result with their edit distance. If the current candidate list has the same counter with the same situation (such as in SNOMED CT or has the same parents). The final result is depended on the edit distance, which the lowest edit distance will be the result (See Figure 2).

The following figure (Figure 1)is the high-level flowchart for the current mapping system. The UIL searcher showing in Figure 3. The Ranking Model in Figure 1 is the bottom of Figure 2.

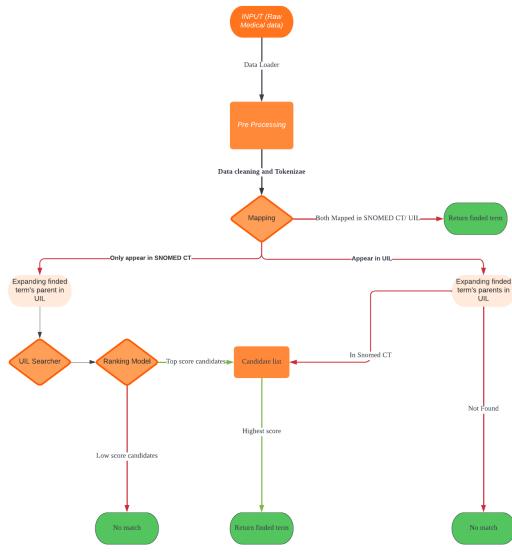


Figure 1

The following is Figure 2 is the final Mapping system's flowchart in this project. There are more details than in Figure 1, but it may be hard to understand, therefore there exists figure 1 to help understand the mapping algorithm.

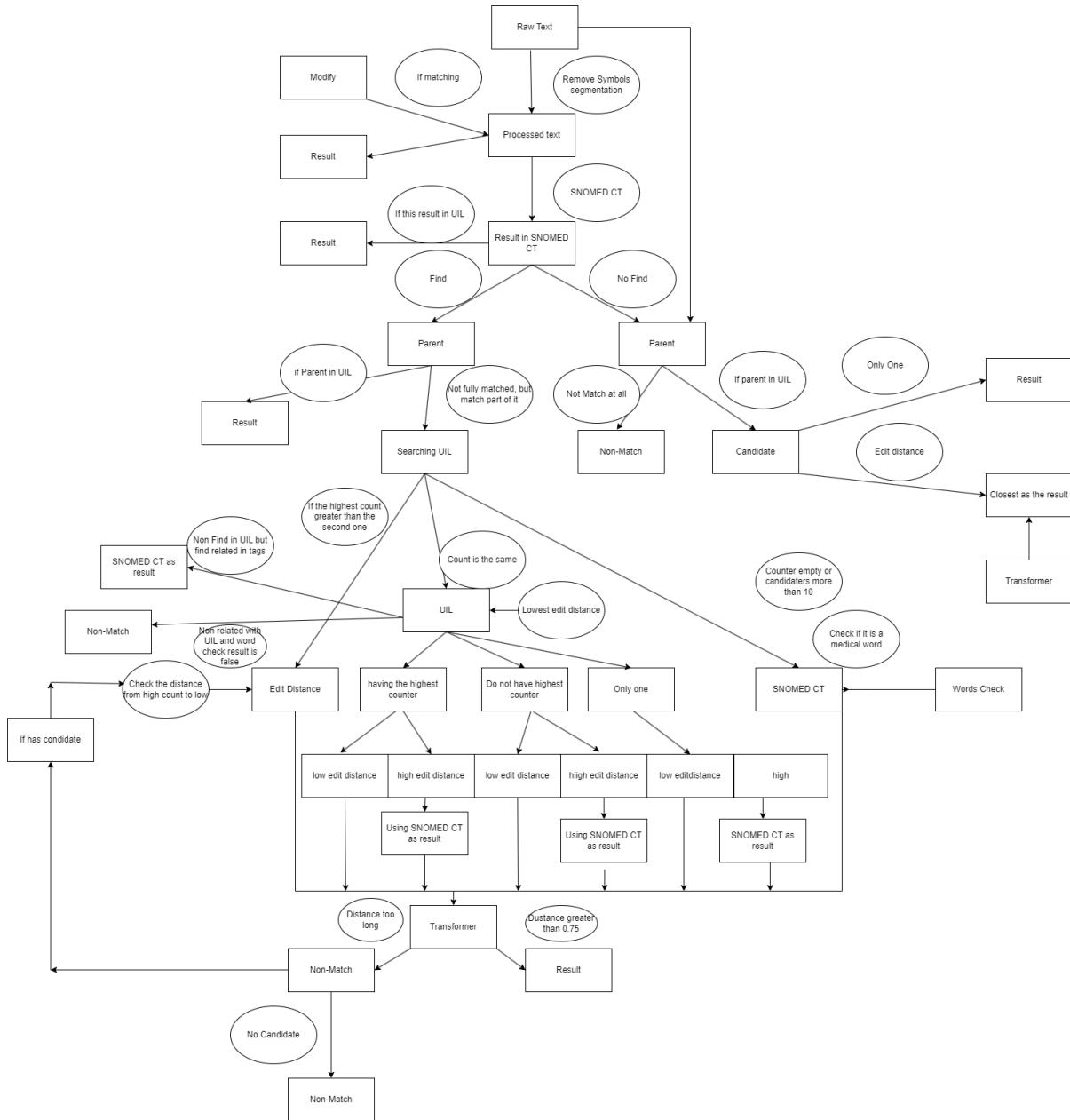


Figure 2

The following is the 'search in UIL' in Figure 2 or 'UIL searcher' in Figure 1. It shows the whole process that how the mapping system works in searching in UIL.

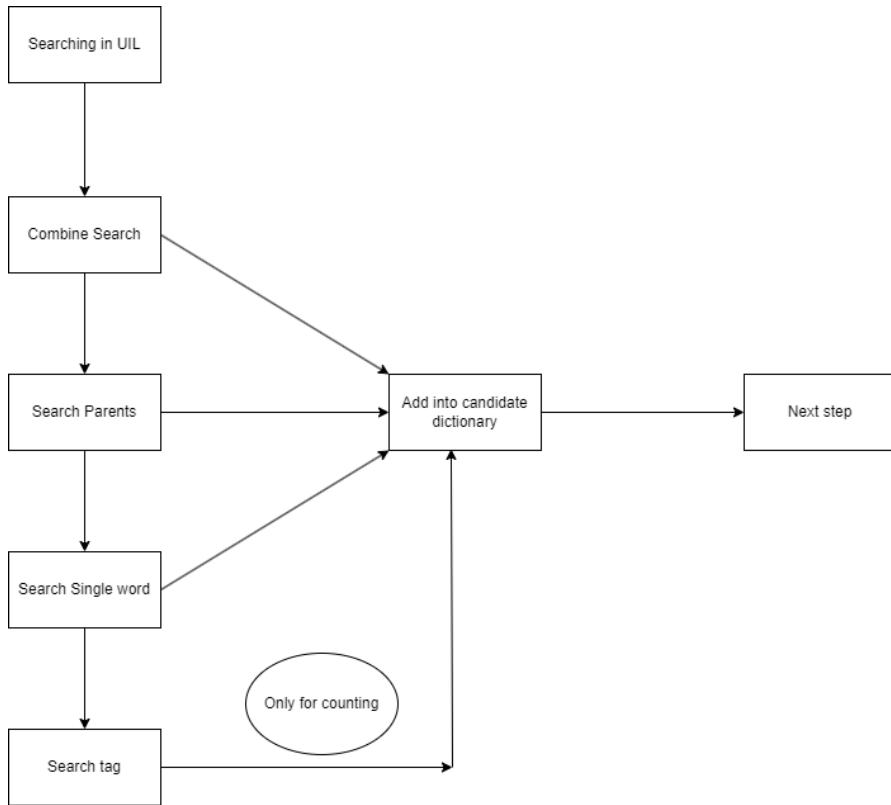


Figure 3

Future Challenge

1. Words Library
 - a. There is not or we do not find a dictionary or library, which only contains medical words so that we can filter for terms that are not related to medicine.
 - i. In the raw text, there are some words entirely unrelated to medical symptoms, such as travel and review.
 - b. There is a medical word-checking list in the word-check system but it is not working perfectly.
2. The UIL is imperfect.
 - a. There are 600 kinds of infection in the UIL, which are many missing. For example, if the SNOMED CT is Sinusitis, but the UIL only contains Chronic sinusitis, Invasive fungal sinusitis, Complicated acute rhinosinusitis and Uncomplicated acute rhinosinusitis.
3. Operating time
 - a. Loading libraries and models takes time.
 - i. The neural network needs time for loading parameters

A full explanation of the mapping algorithm (Figure 2)

Notice: This is the explanation for the whole process, it is very complex.

1. Preprocessing
 - a. UIL
 - i. Convert all uppercase to lowercase
 - b. Raw Text
 - i. Remove all symbols and use special symbols as the basis of the tokenizer.
 - ii. convert all cases to lowercase
2. SNOMED CT Searcher
 - a. Using the processed raw text input SNOMED CT and get the result
 - b. If the result has the same in UIL, then return the result.
 - c. If Find in SNOMED CT but there is not the same one in UIL
 - i. Find the parent
 1. If the parent can be found, and the same in UIL and then return the result
 2. If there is a parent but there is not the same in UIL
 - a. into UIL searcher (Figure 3)
 - i. If there is the highest counter
 1. Into transformer
 - a. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
 - ii. If there is not the highest counter or there is too much edit distance list
 1. Using Edit distance in SNOMED CT and Candidate list
 - a. If there are too many candidate results, using SNOMD CT result

- i. Check in the Transformer
 - 1. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
 - 2. If none of them is good, then use the SNOMED CT result
- b. Using the shortest Edit distance one as the result.
 - i. Check in the transformer
 - ii. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
- b. If there is the same as a parent in UIL
 - i. return the result
 - ii. Into transformer
 - 1. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
- ii. No parents
 - 1. Into UIL searcher (Figure 3)
 - a. combined Search
 - i. If there are multiple results in a combined search
 - 1. Using the closest edit distance
 - 2. Into transformer
 - a. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
 - ii. if there is only one
 - 1. into transformer
 - a. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
 - b. Single search
 - i. if there is only one in this section
 - 1. calculate the cosine distance by the transformer.
 - ii. If there are multiple results in this section
 - 1. Using tag in UIL to calculate the rewards for each the single match
 - a. The highest one as a result goes to the transformer
 - i. If the distance is too far, then chose the highest probably one in the candidate list into transformer again.
 - iii. if there is no Result in this section
 - 1. return 'Non-Match'
 - 3. The rewards calculate
 - a. There are two main parts to UIL, infection and tag
 - i. infection is the final result, therefore if the word has been mentioned here it add to the candidate list with new rewards
 - ii. Tag is a description of the infection, if the candidate list has been mentioned, calculate the new result rewards but do not add to the candidate list.

Mapping Scenario

Scenario 1

Raw text: Sinusitis

Output is 'Chronic sinusitis'

Candidate list: Chronic sinusitis, invasive fungal sinusitis, uncomplicated acute rhinosinusitis and Complicated acute rhinosinusitis.

SNOMED CT result: Sinusitis

Reason: There are the same rewards, and parents and the SNOMED CT are not in UIL, therefore using edit distance to find the closest one, which is 'Chronic sinusitis'.

Scenario 2

Raw text: Constipation

Output is 'Constipation'

Candidate list: None

SNOMED CT result: 'Constipation'

Reason: This is medical work, so the result should be the return of the SNOMED CT because it is not mentioned in UIL.

Scenario 3

Raw text: 'Check up'

Output is Non-Match

Candidate list: None

SNOMED CT result: 'Check'

Reason: Because this is not a medical word, even if it has returned result by SNOMED CT, it still should be 'Non-Match'

Scenario 4

Raw text: 'T2DM'

Output is 'Type 2 Diabetes Mellitus'

Candidate list: None

SNOMED CT result: 'Type 2 diabetes Mellitus'

Reason: UIL's tag has many mentions, but the tag does not have enough support for the mapping result. In addition, the edit distance is very long, but the cosine distance in the transformer is very low. Therefore, it should return the SNOMED CT's result.

Sprint Plan & Review

- Sprint 2 Plan
- Sprint 2 Backlog
- Sprint2 - Review
- Sprint 3 Plan
- Sprint3 - Review
- Sprint 3 Backlog
- Sprint 4 Plan
- Sprint4 - Review
- Sprint 4 Backlog
- Kanban in Trello
- Technology Plan

Kanban in Trello

Trello: <https://trello.com/b/1BfJnJkL/di-redback-kanban>

Status	Version	Date
DONE	3.0	2023-5-24
DONE	2.0	2023-4-27
DONE	1.0	2023-3-23

Sprint3

The Trello board for Sprint 3 is organized into six columns:

- Label Description:** A sidebar containing color-coded labels and their descriptions: 1-2 days workload (green), 3-4 days workload (yellow), 5-7 days or more workload (orange), Functional Debugging (blue), Functional Testing (purple), Development sprint (teal), and UI Design (pink).
- To Do:** Contains one card labeled "To-Do" with a yellow emoji.
- Doing:** Contains one card labeled "Doing" with a green emoji.
- Code Review:** Contains two cards, both labeled "Code Review" with a blue emoji. The first card has a due date of May 23 and the second of May 24.
- Backlog-Sprint3:** Contains one card labeled "Backlog" with a grey emoji.
- Sprint3-Done:** Contains one card labeled "Done" with a red emoji.

Below the columns, there are several user stories listed:

- US18-As a researcher, I want to upload the new list of standardized disease descriptions to the system so that I can get the result I wanted. Set a button on the front end for users to upload standardize the disease description list (SNOMED-CT, UIL) [Due May 26]
- US10-As a researcher, I want to see the distance to other possible results when I see the result. The front end displays the distance from the result to the raw data on each data transmitted by the back end. 4points [Due May 1 - May 4]
- US12- As a researcher, I want to see the upload file state(processing , finish). The front end displays the file processing status according to the file processing status data provided by the back end. 6 Points [Due May 1 - May 4]

Sprint2

Sprint1

Sprint1

The screenshot shows a Trello board titled "DI-Redback Kanban" with three columns: "Backlog", "To Do", and "Doing". The "Doing" column is expanded, showing several user stories (US) and their details:

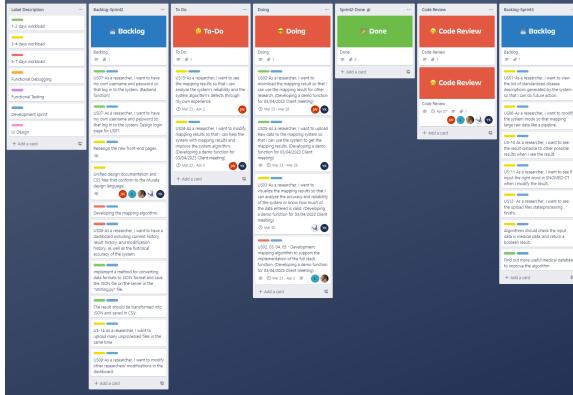
- US01: As a researcher, I want to see the mapping results so that I can analyze the reliability of the system and the defects of the system algorithm through my own experience.
- US02: As a researcher, I want to upload Raw data to the mapping system so that I can use the system to get the mapping results.
- US03: As a researcher, I want to modify the system mode so that mapping large raw data like a pipeline.
- US04: As a researcher, I want to have my own username and password so that log in to the system.
- US05: As a researcher, I want to see the mapping results so that I can analyze the accuracy, reliability, and security of the system.
- US06: As a researcher, I want to curate mapping results so that I can help system to statistical results and improved system algorithm.

A "Sprint checklist" card is also present in the "Doing" column. The "Done" column for Sprint1 is empty.

Sprint2

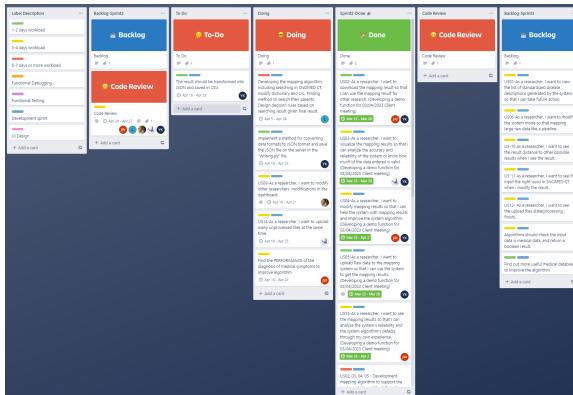
Sprint2 Week5

Allocating the Backlogs of Sprint2, completing the functional instance development before 03/04/2023 Client Meeting.



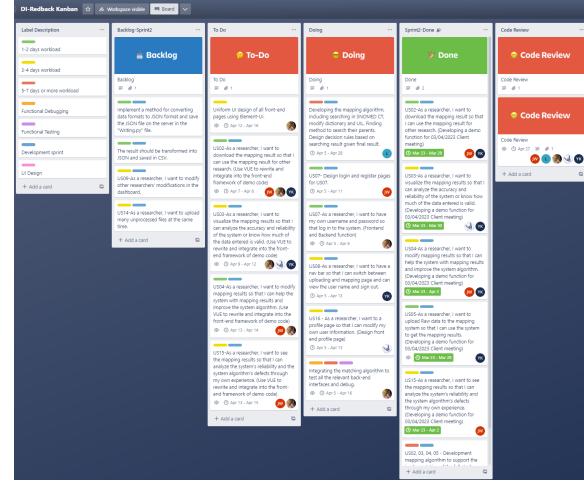
Sprint2 Week7

Modifying some functions based on client feedbacks.



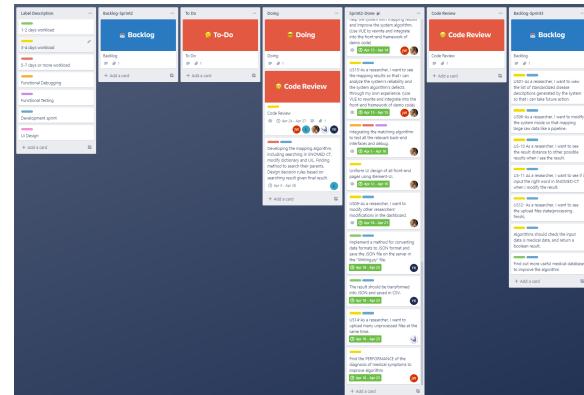
Sprint2 Week6

Based on Week5 demo, the page is redesigned, and all the necessary functions have been developed.



Sprint2 Week8

Code Review



Sprint3

The board includes a sidebar with 'Label Description' and a footer with a 'Done' section.

Label Description:

- 1-2 days workload
- 3-4 days workload
- 5-7 days or more workload
- Functional Debugging
- Functional Testing
- Development sprint
- UI Design

To Do: 1 card (To Do)

Doing: 1 card (Doing)

Code Review: 2 cards (Code Review, Code Review2)

Backlog-Sprint3: 4 cards (Backlog, US18, US11, US10)

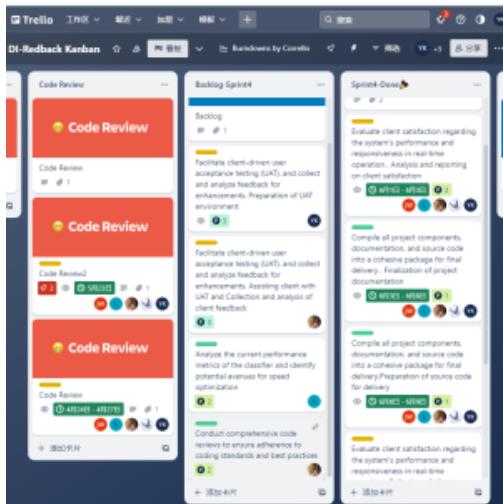
Sprint3-Done: 2 cards (Done, US12)

Done: 2 cards (Done, US11)

Footer: Done (2 cards), US18, US11, US10, US12

Sprint4

Trello: <https://trello.com/b/1BfJnJkL/di-redback-kanban>



Sprint Plan

This page houses documentation related to the sprint plans of Digital Health.

- [Sprint 2 Plan](#)
- [Sprint 3 Plan](#)
- [Sprint 4 Plan](#)

Sprint 2 Plan

Goal

- Prototype development and presentation
- Software Architecture Establishment
- Provide at least 2 options for medical data Mapping algorithms
- Front-end development complete
- Ensures lightweight development and local deployment capabilities

Sprint planning checklist

Preparation	Meetings plan	Follow up
<input checked="" type="checkbox"/> Product Backlog <input checked="" type="checkbox"/> Sprint 2 Backlog <input checked="" type="checkbox"/> Product demostration video <input checked="" type="checkbox"/> Quality and Risk control (risk management, testing) <input checked="" type="checkbox"/> Design and Build the Prototype <input checked="" type="checkbox"/> Establish final Architecture (Front-end, Back-end, Algorithms)	<input checked="" type="checkbox"/> Weekly Client Meeting <input checked="" type="checkbox"/> Stand-up Team Meeting <input checked="" type="checkbox"/> Weekly Supervisor meeting	<input checked="" type="checkbox"/> Subtask on the product backlog and assign these tasks in Trello <input checked="" type="checkbox"/> Requirement update (User stories, persons update) <input checked="" type="checkbox"/> Front-end Design <input checked="" type="checkbox"/> Development details update (Technical analysis , Mapping System)

Previous sprint summary

Sprint theme	Requirement analysis, User story, client meetings, prototype design, Persons
Summary	<ul style="list-style-type: none">• Project requirements are identified• Sprint 2 backlog content identified• A communication plan is defined• Identify specific technology scopes• Successful links to medical services provided by our client

Sprint team members

Name	Role
Yanxi KE	Documentation Manager, developer
Runzhe HUA	Scrum Master, developer
Lang Cao	Product Owner, developer
Jiacheng Wang	Testing Lead, developer
Mingjun GAO	Deployment Lead, developer

Capacity planning

	Current sprint
Total days	37 days
Projected capacity	179 points
Individual capacity	Yanxi KE 27 points Mingjun GAO 42points

Lang Cao 53 points
Runzhe HUA 18 points
JingCheng Wang 24 points

Potential risks

Risk	Mitigation
The accuracy of the algorithm may not meet the customer's requirements in the end	Allocate more time between two people to work on the algorithm.
The use of certain medical databases may be contrary to Australian medical policy.	Ask the client if a relational database is suitable for this project.
The understanding of the project is not very clear.	Do some demos first and then discuss with clients.

Sprint 3 Plan

GOALS

1. Improving the mapping system performance.
2. Successful Docker deployment includes medical API links and a machine learning model backend environment.
3. Back-end development is complete, and data manipulation is consistent with the front-end.
4. The non-medical functions of the program are almost complete, and the developer shifts the task to algorithm development.

Sprint planning checklist

Preparation	Meeting	Follow up
<input checked="" type="checkbox"/> Sprint 2 Backlog <input checked="" type="checkbox"/> Client history feedback <input checked="" type="checkbox"/> Code Review	<input checked="" type="checkbox"/> Weekly Client Meeting <input checked="" type="checkbox"/> Team Stand-Up Meeting <input checked="" type="checkbox"/> Weekly supervisor meeting	<input checked="" type="checkbox"/> Assign backlog tasks in Trello. <input checked="" type="checkbox"/> Code review <input checked="" type="checkbox"/> Requirement update <input checked="" type="checkbox"/> Development update <input checked="" type="checkbox"/> Product backlog update <input checked="" type="checkbox"/> Quality update <input checked="" type="checkbox"/> Product docker update <input checked="" type="checkbox"/> Mapping system update

Previous sprint summary

Sprint theme	Requirement analysis, User story, client meetings, prototype design, Persons, product build
Summary	<ul style="list-style-type: none">Front-end development is substantially complete, some small problems need to be fixedThe back-end medical data and user login registration have been integrated into the front-end functionality.The medical data mapping algorithm shows acceptable performance, but processing speed could be improvedThe data visualization and presentation have been generally satisfactory to the clientThe required environment can be deployed directly via Docker.

Sprint team members

Name	Role
Yanxi KE	document management, developer
Runzhe HUA	Scrum Master, developer
Lang Cao	Product Owner, developer
Jiacheng Wang	Testing Lead, developer
Mingjun GAO	Deployment Lead, developer

Capacity planning

Sprint backlog: [Sprint 3 Backlog](#)

	This sprint	Previous sprint
Total days	28 days	37 days
Projected capacity	139 points	179 points
Individual capacity	Yanxi KE 22 points Mingjun GAO 28points	Yanxi KE 27 points Mingjun GAO 42 points

Lang Cao 30 points	Lang Cao 53 points
Runzhe HUA 17points	Runzhe HUA 18points
JingCheng Wang 23 points	JingCheng Wang 24 points

Potential risks

Risk	Mitigation
The customer's understanding of the algorithm is limited, and the group and the customer may have very different ideas about the algorithm. Ultimately makes the product less effective.	Try to use more meetings and time to explain further and communicate with customers about the algorithm part.
The final product will operate on Windows system servers, but we cannot test it.	Talk about it at the next meeting.

Sprint 4 Plan

GOALS

1. Deploy the project to the customer's production environment.
2. Continue to improve project documentation.
3. Review the past experience of this project.

Sprint planning checklist

Preparation	Meeting	Follow up
<input checked="" type="checkbox"/> Deployment	<input checked="" type="checkbox"/> Weekly Client Meeting <input checked="" type="checkbox"/> Team Stand-Up Meeting <input checked="" type="checkbox"/> Weekly supervisor meeting	<input checked="" type="checkbox"/> Quality update <input checked="" type="checkbox"/> Confluence update <input checked="" type="checkbox"/> Trello update <input checked="" type="checkbox"/> Mapping system update

Previous sprint summary

Sprint theme	Requirement analysis, User story, client meetings, prototype design, Persons, product build
Summary	<ul style="list-style-type: none">• Improving the mapping system performance.• Successful Docker deployment includes medical API links and a machine learning model backend environment.• Back-end development is complete, and data manipulation is consistent with the front-end.• The non-medical functions of the program are almost complete, and the developer shifts the task to algorithm development.• Demo product ppt and corresponding presentation• Maintain iterative products

Sprint team members

Name	Role
Yanxi KE	document management, developer
Runzhe HUA	Scrum Master, developer
Lang Cao	Product Owner, developer
Jiacheng Wang	Testing Lead, developer
Mingjun GAO	Deployment Lead, developer

Capacity planning

Sprint backlog:

	This sprint	Previous sprint
Total days	12 days	28 days
Projected capacity	14 points	139 points
Individual capacity	Yanxi KE 3 points Mingjun GAO 3 points Lang Cao 3 points Runzhe HUA 4 points JingCheng Wang 4 points	Yanxi KE 22 points Mingjun GAO 28 points Lang Cao 30 points Runzhe HUA 17 points JingCheng Wang 23 points

Potential risks

Risk	Mitigation
Unexpected errors may occur when deploying products due to different deployment environments.	Fully test before delivering the product.

Sprint Backlog

- [Sprint 2 Backlog](#)
- [Sprint 3 Backlog](#)
- [Sprint 4 Backlog](#)

Sprint 2 Backlog

Story Points are based on workload.

Product Backlog: Product Backlog

Trello: Trello

content	SubTask	Is finished	Start Time	Allocate Time (day)	actual time spent (day)	End Time	Points	responsible person
US02-As a researcher, I want to download the mapping result so that I can use the mapping result for other research. (Developing a demo function for 03/04/2023 Client meeting)	Add a download button when finished processing. The files are delivered by the backend for users to download	Yes	23/03	3	3	27/03	3	Jiacheng Wang
	The backend provides the corresponding files to the front according to the request of the frontend	Yes	27/03	2	2	28/03	2	Yanxi Ke
US03-As a researcher, I want to visualize the mapping results so that I can analyze the accuracy and reliability of the system or know how much of the data entered is valid. (Developing a demo function for 03/04/2023 Client meeting)	Visualize the results on the page based on the processed data provided by the backend.	Yes	23/03	3	4	27/03	3	Runzhe hua
	The backend provides corresponding data according to the frontend request	Yes	27/03	3	3	30/03	3	Yanxi Ke
US04-As a researcher, I want to modify mapping results so that I can help the system with mapping results and improve the system algorithm. (Developing a demo function for 03/04/2023 Client meeting)	A window is displayed on the front-end data display page for users to modify data	Yes	27/03	3	4	31/03	3	Jiacheng Wang,
	The backend changes the data corresponding to the processed file on the server according to the modification provided by the frontend.	Yes	30/03	3	3	02/04	3	Yanxi Ke
US05-As a researcher, I want to upload Raw data to the mapping system so that I can use the system to get the mapping results. (Developing a demo function for 03/04/2023 Client meeting)	The front end adds an upload button and an upload area for users to upload pending files	Yes	02/04	3	2	04/04	3	Yanxi Ke
	The backend saves the files on the server according to the files uploaded by the front end and processes the files through the algorithm interface. Save the resulting file to the server.	Yes	02/04	3	2	04/04	3	Yanxi Ke
US15-As a researcher, I want to see the mapping results so that I can analyze the system's reliability and the system algorithm's defects through my own experience. (Developing a demo function for 03/04/2023 Client meeting)	The front end displays the processed data provided by the back end on the page	Yes	23/03	3	4	02/04	3	Jiacheng Wang
	The backend provides the corresponding processed data according to the requirements of the frontend	Yes	02/04	3	3	03/04	3	Yanxi Ke
US02, 03, 04, 05 - Development mapping algorithm to support the implementation of the full stack function. (Developing a demo function for 03/04/2023 Client meeting)	Deploy ontoserver to the project docker, and use the Ontoserver database to process input data.	Yes	23/03	3	4	27/03	3	Mingjun Gao
	Download and deploy the scispacy database to the project, and use the scispacy database to process the input data.	Yes	23/03	3	5	26/04	7	Lang Cao
	Design search algorithm and structure, apply Ontoserver database and scispacy database processing algorithm.	Yes	28/03	7	8	03/04	14	Lang Cao
US07- Design login and register pages for US07.		Yes	05/04	7	7	11/04	5	Jiangcheng Wang
US07-As a researcher, I want to have my own username and password so that log in to the system. (Frontend and Backend function)	Front-end implementation of the login page	Yes	05/04	2	2	06/04	3	Mingjun Gao
	The backend processes login information according to the MySQL database	Yes	05/04	2	2	06/04	2	Mingjun Gao
	Connect to MySQL database	Yes	05/04	2	2	06/04	2	Mingjun Gao
US08-As a researcher, I want to have a nav bar so that I can switch between uploading and mapping page and can view the user name and sign out.	Realize the front-end bar function. so that it can jump to each page	Yes	05/04	4	4	09/04	4	Yanxi Ke
	Realization of user bar and realization of logout function	Yes	09/04	3	1	12/03	3	Mingjun Gao
US02-As a researcher, I want to download the mapping result so that I can use the mapping result for other research.	Use VUE to rewrite and integrate into the front-end framework of demo code	Yes	07/04	1	1	08/04	2	Mingjun Gao
US03-As a researcher, I want to visualize the mapping results so that I can analyze the accuracy and reliability of the system or know how much of the data entered is valid.	Use VUE to rewrite and integrate into the front-end framework of demo code	Yes	09/04	3	1	12/04	3	Mingjun Gao

US16 - As a researcher, I want to a profile page so that I can modify my own user information. (Design front end profile page)	Design User Profile Pages	Yes	05/04	3	3	08/04	3	Runzhe hua
	Front-end implementation of user profile page	Yes	08/04	4	4	11/04	4	Runzhe hua
	Back-end processing front-end user personal information changes	Yes	11/04	3	3	13/04	3	Runzhe hua
US04-As a researcher, I want to modify mapping results so that I can help the system with mapping results and improve the system algorithm.	Use VUE to rewrite and integrate into the front-end framework of demo code	Yes	13/04	3	1	14/04	3	Mingjun Gao
US15-As a researcher, I want to see the mapping results so that I can analyze the system's reliability and the system algorithm's defects through my own experience.	Use VUE to rewrite and integrate into the front-end framework of demo code	Yes	13/04	5	2	15/04	5	Mingjun Gao
Integrating the matching algorithm to test all the relevant back-end interfaces and debug.		Yes	05/04	10	11	16/04	5	Mingjun Gao
Uniform UI design of all front-end pages using Element-UI.		Yes	12/04	7	4	16/04	5	Mingjun Gao
US09-As a researcher, I want to modify other researchers' modifications in the dashboard.	Display all user submissions and processing results on the mapping page	Yes	18/04	3	1	21/04	3	Mingjun Gao
Find the PERFORMANCE of the diagnosis of medical symptoms to improve algorithm		Yes	18/04	4	5	23/04	2	Jiacheng Wang
Implementing a machine learning classifier for determining whether incoming data matches medical text		Yes	24/04	6	6	30/04	5	Jiacheng Wang
Implement a method for converting data formats to JSON format and save the JSON file on the server in the "Writing.py" file.		Yes	18/04	5	6	23/04	3	Yanxi Ke
The result should be transformed into JSON and saved in Json.		Yes	18/04	3	4	23/04	3	Yanxi Ke
US14-As a researcher, I want to upload many unprocessed files at the same time.		Yes	23/04	6	5	23/04	5	Runzhe Hua
Developing the mapping algorithm, including searching in SNOMED CT, modify dictionary and UIL. Finding method to search their parents. Design decision rules based on searching result given final result.	Download and deploy the mpi4py database to the project.	Yes	30/04	7	6	05/04	7	Lang Cao
	Use the transformer model to judge the correctness of the result and modify the processed result according to the correctness	Yes	30/04	15	14	28/04	25	Lang Cao
	Display the distance from the result to the possible result according to the algorithm operation result	Yes	30/04	3	1	28/04	3	Mingjun Gao

Sprint 3 Backlog

Content	SubTask	Is finished	Start Time	Allocate Time (day)	actual time spent (day)	End Time	Points	responsible person	Note
US01-As a researcher, I want to view the list of standardized disease descriptions generated by the system so that I can take future action.	Set a button on the front end for users to download and re-standardize the disease description list (SNOMED-CT, UIL)	Yes	28 April 2023	5	23	21 May 2023	3	Lang Cao	
	The backend returns the corresponding SNOMED-CT and UIL files according to the frontend requirements.	Yes	28 April 2023	5	23	21 May 2023	3	MingJun Gao	
US06-As a researcher, I want to modify the system mode so that mapping large raw data is like a pipeline.	Frontend setup with a button or page transition system.	Yes	30 April 2023	4	22	21 May 2023	3	MingJun Gao	Client agreed with two modes work the same way
US-10 As a researcher, I want to see the result distance to other possible results when I see the result.	The front end displays the distance from the result to the raw data on each data according to the data transmitted by the back end.	Yes	30 April 2023	4	4	4 May 2023	4	Lang Cao	
	The distance between the result returned by the backend and the raw data	Yes	30 April 2023	4	4	4 May 2023	4	Lang Cao	
US-11 As a researcher, I want to see if I input the right word in SNOMED-CT when I modify the result.	The backend returns possible modification results, and at the same time, when the front end makes modifications, it is passed to the backend for judgment.	Yes	1 May 2023	3	4	4 May 2023	4	Lang Cao /MingJun Gao	
	The front end displays a window for users to modify data and generates a drop-down box to provide possible modification items.	Yes	1 May 2023	3	3	4 May 2023	4	Lang Cao /MingJun Gao	
US12- As a researcher, I want to see the upload files state(processing , finish).	The front end displays the file processing status according to the file processing status data provided by the back end.	Yes	1 May 2023	3	3	4 May 2023	6	MingJun Gao	
	The backend returns the processing file status.	Yes	1 May 2023	3	3	4 May 2023	6	MingJun Gao	
Algorithms should check the input data is medical data and return a boolean result.		Yes	28 April 2023	4	5	23 May 2023	6	Jiacheng Wang	
Find out more useful medical databases to improve the classification algorithm.		Yes	6 May 2023	16	17	23 May 2023	6	Jiacheng Wang	
Integrate the installation process into the Bat file.		Yes	16 May 2023	10	9	25 May 2023	12	Yanxi Ke /Runzhe Hua	
Algorithm introduction file (detailed steps and examples)		Yes	16 May 2023	6	5	21 May 2023	8	Lang Cao	
presentation file preparation		Yes	16 May 2023	5	9	25 May 2023	30	Team	
ChatGPT Review		Yes	6 May 2023	15	15	21 May 2023	10	Yanxi Ke	
US18-As a researcher, I want to upload the new list of standardized disease descriptions to the system so that I can get the result i wanted.	Set a button on the front end for users to upload standardize the disease description list (SNOMED-CT, UIL)	Not	16 May 2023	4	5	21 May 2023	5	Runzhe Hua	
	The backend saves the corresponding SNOMED-CT and UIL files in the server.	NOT	16 May 2023	4	5	21 May 2023	5	Jiacheng Wang	

Sprint 4 Backlog

Content	SubTask	Is finished	Start Time	Allocate Time (day)	actual time spent (day)	End Time	Points	responsible person	Note
Perform compatibility testing of the project on the client's designated environment.	Execution of compatibility tests	Yes	1 June	5	6	6 June	3	Jiacheng Wang	
	Recording and analysis of test results	Yes	1 June	5	6	6 June	3	Runzhe Hua	
Evaluate client satisfaction regarding the system's performance and responsiveness in real-time operation.	Collection of client feedback	Yes	30 May	1	1	31 May	2	Lang Cao	
	Analysis and reporting on client satisfaction	Yes	1 June	2	2	3 June	2	Yanxi Ke	
Conduct comprehensive code reviews to ensure adherence to coding standards and best practices.		Yes	6 June	2	2	8 June	2	Mingjun Gao	
Compile all project components, documentation, and source code into a cohesive package for final delivery.	Finalization of project documentation	Yes	3 June	3	6	8 June	1	Team	
	Preparation of source code for delivery	Yes	8 June	1	1	8 June	1	Team	

Sprint Review

[Sprint2 - Review](#)

[Sprint3 - Review](#)

[Sprint4 - Review](#)

Sprint2 - Review

Team Review Meeting: 2023-4-26

Meeting link here [26/04/2023 Team Meeting](#)

Sprint 2 Goal:

During this sprint, we aimed to establish a system that supports users to upload data, download mapping results, view and modify the mapping and has a modern interface. We have essentially achieved this objective.

Sprint 2 Backlog:

We completed the following items from our product backlog during this sprint.

Demo:

This video shows the working product increment that was produced during the sprint.

Feedback

During this sprint, we received feedback from our clients on the modifications that were made to the system. They were pleased with the progress that was made and the fact that we could implement the changes they requested. They also provided valuable feedback on the visualization and data modification functions, which we have taken into account for future sprints.

Product increment

Overall, this sprint was a success. We were able to complete all the items on our product backlog and we received positive feedback from our clients. However, we did encounter some challenges. In future sprints, we will need to allocate more time to these areas to ensure that they are completed in a timely manner. Additionally, we will need to work on improving the system's performance to handle larger datasets.

On the other hand, due to the negligence in team organization, team management, and team communication at the beginning of Sprint 1, there were many defects in Sprint 1 and we did not receive a high score. During the mid-term of Sprint 2, we organized a special internal team meeting to discuss these issues and provide a platform for all team members to express their genuine thoughts. Based on this, we re-established a communication management platform and implemented new management measures. In Sprint 3, we will continue to adopt this new management and communication approach and continue to iterate on the methods, thought processes, and outcomes in this regard.

On the other hand, due to the initial negligence in team organization, management, and communication in Sprint 1, there were many defects, resulting in a low score. In the middle of Sprint 2, we organized a dedicated internal team meeting to discuss these issues and provide a platform for team members to express their true feelings. Based on this, we rebuilt our communication management platform and implemented new management methods.

Group Efficiency

In this sprint cycle, we completed a total of 300 points of tasks within 28 days. The average is 8.1 points/day. The overall team efficiency is very good. I hope the next sprint will continue.



Next Steps:

In sprint 3. Our team will continue improving the mapping system's accuracy and calculation speed. Find a new medical thesaurus and simultaneously study how to deploy the entire system on docker so users can use it more easily. We will continue to use this new management and communication approach, further iterating on the methods and reflecting on our processes and results.

Sprint3 - Review

Team Review Meeting: 2023-5-26

Meeting link here [26/05/2023 Team Meeting](#)

Sprint 3 Goal:

Our primary goals for Sprint 3 focused on enhancing the mapping system performance and Docker deployment. We have achieved this objective.

Sprint 3 Backlog:

We completed these items from our product backlog during this sprint.

Demo:

[These videos show the working product increment that was produced during the sprint.](#)

Feedback

Our customers are pleased with the overall progress made. We have revised the flow chat about the principle according to the customer's request to make it easier to understand.

Product Increment

Overall, this sprint was a success. We were able to complete all the items on our product backlog, and we received positive feedback from our clients. We made some strides in enhancing our system's features. One of the main achievements was allowing researchers to view and manipulate a standardized disease descriptions list. This was realized through a frontend button for downloading and re-standardizing the disease list, with the backend designed to return and save the appropriate files. Moreover, we improved the system mode to facilitate large raw data mapping akin to a pipeline and enabled the system to display the result distance to other possible outcomes. We added a verification function in the modification part and implemented a system to display the upload file's status. To ensure data integrity, our algorithms were tailored to confirm if the input data is medical. Lastly, we streamlined the installation process via a Bat file. The completion of these allows us to meet the goals of the entire project basically.

Group Efficiency

In this sprint cycle, we completed a total of 151 points of tasks within 28 days. The average is 5.39 points/day. Compared with the 8.1 points/day of the last sprint. This is because most of the development work is completed in sprint2, and sprint3 is more about new requirements from customers and iterative maintenance of the original version. We ourselves are very satisfied with the progress of the project development. In the beginning, because most of the other courses of the team members have reached the AS3 period, the progress is relatively slow. Subsequent team members will have more time and efficiency to complete the work of sprint 3 as their free time increases.



Next Steps:

Our next step will be to continue reflecting on our experiences and lessons learned and modify our work based on the feedback from this sprint. We aim to wrap up the project and deliver it during the next sprint, including complete project documentation and the algorithm description. Our sprint 4 primary objective is to complete and deliver the entire project.

Sprint4 - Review

Team Review Meeting: 2023-6-7

Meeting link here [07/06/2023 Team Meeting](#)

Sprint 4 Goal:

- **Package and Release Stable Version:** Integrate all project components, followed by the release of a tested and stable version. This ensures functionality while maintaining the reliability of the software.
- **Code Optimization and Documentation:** Improve the codebase for better readability and maintainability. The accompanying documentation is prepared.
- **System Performance Optimization:** Enhance the performance of the existing systems, potentially reducing execution time, increasing accuracy, and minimizing resource consumption.
- **User Documentation Completion:** Refine the user manual, introducing more detailed instructions, comprehensive feature descriptions, and intuitive examples. The end product should be a comprehensible and thorough document that assists users in maximizing product utility.

Sprint 4 Backlog:

During this sprint, we completed all items in the product backlog.

Demo:

These videos show the working product increment that was produced during the sprint.

Feedback

1. **Overall Project Satisfaction and Future Development:** Customers have expressed overall satisfaction with the current state of the project. They have indicated interest in future development, suggesting confidence in the project's progress and direction.
2. **Deployment Testing:** Customers will conduct a real-world deployment test after obtaining deployment permissions from the school server.
3. **Documentation and Code Status:** Customers believe the documentation could be more comprehensive. And they are satisfied with the codebase, indicating no further changes are required at this stage.

Product Increment

In this Sprint, the primary focus has been the finalization and optimization of the project. The collective efforts of the team have been deployed to align all features with the expected standards. Comprehensive final tests have been executed to ascertain software quality and user satisfaction. This includes meticulous attention to detail in refining the feature set, code optimization for performance and readability, as well as augmenting the documentation for a more user-friendly experience. The end product is a more polished, efficient, and user-oriented software solution, demonstrating an appreciable increment from the previous state.

Technology Plan

	Sprint2	Sprint3
Project Doc	Confluence	Confluence
Backlog Kanban	Trello	Trello
API	ontoserver	ontoserver
Front-End	HTML CSS Javascript Vue React	HTML CSS Javascript Vue React
Back-End	Flask	Flask
DB	Mysql	Mysql
VM	docker	docker
UI	element-ui	element-ui
Back-End Test	Postman	Postman

Quality

- [Code Review](#)
- [Testing & Evaluation](#)
- [Coding Standards](#)
- [Risk Management](#)

Code Review

Our review process was conducted meticulously through a combination of careful manual inspection by our team.

Please take a look at each of our code review pages for specific details and insights.

This code review is based on the `sprint2_milestone2` branch on GitHub and focuses on the core aspects of the code.

[Front-end review](#)

[Back-end Review](#)

[Algorithm Review](#)

Algorithm Review

Medical data classifier review

Reviewer: Lang Cao

Developer: Jiacheng Wang

Medical data classifier

```
#Description: This file is used to train the model and predict the label for the medical data, to classify whether the input raw text is legit medical text or not.

import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, confusion_matrix, f1_score

# Train the model and a simple evaluation
def train_nb_classifier():
    train_data = pd.read_csv('MedicalDataClassifier/Medical_data/med_cls_train.csv')

    #split the data into training and testing sets
    X_train, X_test, y_train, y_test = train_test_split(
        train_data['text'], train_data['label'], test_size=0.1, random_state=42
    )

    #vectorize the text first
    vectorizer = CountVectorizer()
    X_train_vec = vectorizer.fit_transform(X_train)
    X_test_vec = vectorizer.transform(X_test)

    #using the Multinomial Naive Bayes classifier
    clf = MultinomialNB()
    clf.fit(X_train_vec, y_train)

    y_pred = clf.predict(X_test_vec)
    accuracy = accuracy_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred, average='weighted')
    print('Accuracy: ', accuracy)
    print('F1 score: ', f1)

    return clf, vectorizer

# Predict the label for a raw text
def predict_medical_text(clf, vectorizer, raw_text):
    raw_text_vec = vectorizer.transform([raw_text])
    label = clf.predict(raw_text_vec)
    return bool(label[0])

# Usage example
if __name__ == '__main__':
    classifier, text_vectorizer = train_nb_classifier()
    test_text = "[ 'heart', 'pain' ]"
    result = predict_medical_text(classifier, text_vectorizer, test_text)
```

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or feedback
AIS1.1	Using the 'weighted'	Use 'macro' or 'micro' average options in f1_score(). It provides a better	Y	Tested, the function returns all the weighted,

	average in F1 score calculation	understanding of the classifier's performance across different classes.		macro and micro F1 scores correctly.
AIS1.2	Using CountVectorizer	Consider using TfidfVectorizer instead of CountVectorizer, as it takes into account both term frequency and inverse document frequency, which can result in better performance.	Y	Tested. TfidfVectorizer does show the ability to make NB more focus on the key words, the accuracy is slightly improved.
AIS1.3	Limited evaluation metrics	Consider adding other evaluation metrics like precision, recall, and confusion matrix to provide more insights into the classifier's performance.	Y	Tested. Add a separate function "evaluate_model", with Recall, Precision, Confusion matrix, and different types of F1 scores.

(Solved in GitHub commit '[update for medical](#)' classifier in branch 'sprint2_lang' for future development. Not in sprint2_milestone2)

Mapping Algorithm Core Function Review

Reviewer: Jiacheng Wang

Developer: Lang Cao

Core Mapping Function

```
def mapping(self):
    ct_result = snomed_ct_dict(self.file)
    finding_id = 0
    # autoModel(self.mod_dict,self.Non_process_text)
    for finding in self.raw_plain.keys():
        # print(finding)
        self.result_dict = []
        # curr_ct = ct_result[left_mapping_text]
        finding_id += 1
        # print(finding)
        self.statement_check = False
        self.check_ct = False
        pre_data = self.raw_plain[finding]['processed']
        self.result_dict = {}
        self.target = ""
        self.ct_find = False
        self.ct_available_check = False

        left_mapping_text = self.Non_process_text[self.raw_text_counter]
        curr_ct = ct_result[left_mapping_text]
        self.raw_text_counter += 1
        self.spacy_pos(left_mapping_text)
        result = ""
        basic_length = 2
        tmp_parent = ""

        for simgle_word in self.parent:
            tmp_parent += simgle_word.lower()

        # print(curr_ct)
        if curr_ct != "Not Find":
            self.ct_find = True
            self.ct_search(curr_ct)
            if self.result_dict:
                for rr in self.result_dict.keys():
                    self.result_mapping.append([left_mapping_text, rr, "SNOMED CT"])
                    self.check_ct = True
                    break

        # modify
        if finding in self.mod_dict.keys():
            self.result_mapping.append([left_mapping_text, self.mod_dict[finding], "UIL"])

    elif self.check_ct == False:
        self.parent = tmp_parent
        if self.ct_find:
            self.find_parent(curr_ct)
```

```

else:
    self.find_parent(self.parent)
if len(pre_data) >= 2:
    for i in range(len(pre_data)):
        if i + basic_length <= len(pre_data):
            tmp_finding = pre_data[i:i+basic_length]
            tmp_string = ""
            for each_in_combine in tmp_finding:
                tmp_string += str(each_in_combine[0])
                tmp_string += " "
            self.find_comb(tmp_string[:-1])
if self.ct_find:
    for ct_tmp in re.split(' ', curr_ct):
        tmp_low = ""
        for i_low in ct_tmp:
            tmp_low += i_low.lower()
        self.target=tmp_low
        # print(self.target)
        self.search(2)
else:
    for target in pre_data:
        # print(target)
        self.target = target[0]
        self.search(1)
# print(Counter(self.result_dict))
# print(self.ct_find)
# print(self.result_mapping)
if self.statement_check:

    # best, score = autoModel(left_mapping_text, self.result_dict)
    # print(best)
    # print(score)
    # if score >= 0.85:
    #     self.result_mapping.append([left_mapping_text, best, "UIL"])
    # else:
    #     self.result_mapping.append([left_mapping_text, "Non-Match", "UIL"])
    # highest uil finding
    if len(self.result_dict)>= 2 and Counter(self.result_dict).most_common(2)[0][1] != Counter(self.result_dict).most_common(2)[1][1]:
        self.result_mapping.append([left_mapping_text, Counter(self.result_dict).most_common(1)[0][0], "UIL"])
    # print("highest uil finding")

    # finding lots of in uil also finding in snomed ct
    elif len(Counter(self.result_dict)) >=4 and self.ct_find:
        self.result_mapping.append([left_mapping_text, curr_ct, "SNOMED CT"])
        # print("finding lots of in uil also finding in snomed ct")

    # lots of matched but not snomed ct
    elif len(self.result_dict) > 2 and Counter(self.result_dict).most_common(2)[0][1] == Counter(self.result_dict).most_common(2)[1][1] and not self.ct_find:
        self.result_mapping.append([left_mapping_text, "Non-Match", "UIL"])
        # print("lots of matched but not snomed ct")

    # find on uil but not find in snomed ct
    elif len(self.result_dict) > 0 and self.ct_find == False:
        n, s = autoModel([left_mapping_text, Counter(self.result_dict).most_common(1)[0][0]])
        # print(s)
        if float(s) > 0.5:
            self.result_mapping.append([left_mapping_text, Counter(self.result_dict).most_common(1)[0][0], "UIL"])
        else:
            self.result_mapping.append([left_mapping_text, "Non-Match", "UIL"])
    # print(Counter(self.result_dict).most_common(2)[0][1])
    # print("find on uil")

    # find in snomed ct
    elif self.ct_find:
        self.result_mapping.append([left_mapping_text, curr_ct, "SNOMED CT"])
        # print("find in snomed ct")
else:

```

```

        self.result_mapping.append([left_mapping_text, "Non-Match", "UIL"])

    # find in snomed ct but not relationship with uil
    elif self.ct_find and self.ct_available_check:
        # print(curr_ct)
        self.result_mapping.append([left_mapping_text, curr_ct, "SNOMED CT"])
        # print("find in snomed ct but relationship with uil")
    else:
        result = "Non-Match"
        self.result_mapping.append([left_mapping_text, result, "UIL"])

    return self.writing.writing(self.result_mapping, self.write_file_name, self.pending_check)
# return self.writing.writing(self.result_mapping, self.write_file_name)

# return self.result_mapping

```

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or feedback
AIS2.1	The function lacks comments and docstrings	Add comments and docstrings to improve code readability and maintainability	Y	Yes
AIS2.2	The function is very long and complex	Consider refactoring the function into smaller and more manageable functions	N	will solve this at the end of the project
AIS2.3	The function has multiple repeated code blocks	Extract repeated code blocks into functions to improve code reusability	N	This is for future implement
AIS2.4	The function performs many unrelated tasks (e.g., searching for concepts in the text, mapping concepts to codes)	Consider breaking the function down into smaller functions that each perform a single task	P	
AIS2.5	The function contains some unused code (e.g., the commented-out automodel code)	Remove unused code to improve code clarity	Y	making it as comment
AIS2.6	The function contains some redundant code (e.g., the if len(self.result_dict) > 0 block could be merged with the first if block)	Simplify code to improve code readability and maintainability	P	This is for future improvement

(Solved in commit '[Update matching AIS2.1-2.6](#)')

Back-end Review

Data Reading Review

Reviewer: Jiacheng Wang
Developer: Lang Cao & Yanxi Ke

Read_Data.py

```
import re
import pandas as pd
import copy
import csv

class read_data:
    def __init__(self):
        pass
    def read_raw(self, file):
        example = {}
        with open(file, 'r') as ex:
            while True:
                current_line = ex.readline().strip()
                # print(current_line)
                count_bit = 0
                for i in current_line:
                    count_bit+=1
                    if i == '\t':
                        break

                line = current_line[count_bit:]
                # print(line)
                if not line:
                    break
                all_line = re.split('-|_|/| ', line)
                tmp_line = []
                for i in range(len(all_line)):
                    new_line = ""
                    for each_alp in all_line[i]:
                        if ('a' <= each_alp <= 'z' or 'A' <= each_alp <= 'Z' or each_alp == ' ' or
                            '0'<each_alp<'9') and each_alp != ' ':
                            new_line += each_alp.lower()

                    if new_line:
                        tmp_line.append([new_line])
                example[line] = {'processed': tmp_line}
        return example

    def read_return_raw(self, file):
        example = []
        with open(file, 'r') as ex:
            while True:
                current_line = ex.readline().strip()
                if not current_line:
                    break
                count_bit = 0
                for i in current_line:
                    count_bit+=1
                    if i == '\t':
                        break

                line = current_line[count_bit:]
                example.append(line)

        return example

    def read_uil_list(self):
```

```

df = pd.read_excel('uil.xlsx')
data=df.values
for i in range(len(data)):
    str1 = ""
    str2 = ""
    str3 = ""
    str4 = ""
    for word1 in data[i][0]:
        str1 += word1.lower()
    data[i][0] = str1

    for word1 in data[i][1]:
        str2 += word1.lower()
    data[i][1] = str2

    for word1 in data[i][2]:
        str3 += word1.lower()
    data[i][2] = str3

    for word1 in data[i][4]:
        str4 += word1.lower()
    data[i][4] = str4

# print(data)
return data

def read_comp(self):
    df = pd.read_excel('human_match.xlsx', header=None)
    data=df.values
    # print(data[0][3])
    return data

def read_his(self):
    result_dic = {}
    csv_reader = csv.reader(open('../modify.csv'))
    for line in csv_reader:
        result_dic[line[0]] = line[1]

    return result_dic

def read_tmp_ct(self, file):
    result_dic = {}
    csv_reader = csv.reader(open(file))
    for line in csv_reader:
        result_dic[line[0]] = line[1]

    return result_dic

```

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or feedback?
BIS1.1	The <code>read_raw</code> method in <code>read_data</code> class uses regex and may slow down processing	Consider using string splitting instead of regex for faster processing	P	Added compatibility for possibly more different formats in the future
BIS1.2	The <code>read_uil_list</code> method in <code>read_data</code> class hardcodes the file name 'uil.xlsx'	Pass the file name as a parameter to make the method more flexible	N	Will add an updated UIL function in the future
BIS1.3	The <code>read_comp</code> method in <code>read_data</code> class hardcodes the file name 'human_match.xlsx'	Pass the file name as a parameter to make the method more flexible	Y	This function can be removed
BIS1.4	The <code>read_tmp_ct</code> method in <code>read_data</code> class hardcodes the file name and assumes the file has only 2 columns	Pass the file name and the number of columns as parameters to make the method more flexible	P	This is for subprocess in MPI. It will delete after reading this file

(Solved in commit '[Update reading data for BIS1.1-1.4](#)')

Snomed CT connector Review:

Reviewer: Jiacheng Wang
Developer: Lang Cao

Snomed CT Connector

```
import warnings
import contextlib
import requests
from urllib3.exceptions import InsecureRequestWarning
from read_data import read_data
from mpi4py import MPI
comm = MPI.COMM_WORLD
rank = comm.Get_rank()
size = comm.Get_size()
import time
start_time = time.time()
import csv
class snomed:
    def __init__(self):
        self.name = ""
        self.url = "https://localhost:8443/fhir/ValueSet/$expand?url=http://snomed.info/sct?fhir_vs=refset
/32570071000036102&count=10&filter="
        self.payload={}
        self.headers = {
            'Accept': 'application/fhir+json',
            'Content-Type': 'application/fhir+json'
        }

    def ct_search(self):
        url = self.url + self.name
        old_merge_environment_settings = requests.Session.merge_environment_settings
        @contextlib.contextmanager
        def no_ssl_verification():
            opened_adapters = set()
            def merge_environment_settings(self, url, proxies, stream, verify, cert):
                opened_adapters.add(self.get_adapter(url))
                settings = old_merge_environment_settings(self, url, proxies, stream, verify, cert)
                settings['verify'] = False
                return settings
            requests.Session.merge_environment_settings = merge_environment_settings
            try:
                with warnings.catch_warnings():
                    warnings.simplefilter('ignore', InsecureRequestWarning)
                    yield
            finally:
                requests.Session.merge_environment_settings = old_merge_environment_settings
                for adapter in opened_adapters:
                    try:
                        adapter.close()
                    except:
                        pass
            with no_ssl_verification():
                response = requests.get(url)
                if "contains" in response.json()["expansion"].keys():
                    return response.json()["expansion"]["contains"][0]['display']
                else:
                    return "Not Find"

        def ct_string_process(self, pre_data):
            string_for_ct = ""
            for ct_i in pre_data:
                string_for_ct += ct_i[0]
                string_for_ct += " "
            self.name = string_for_ct[:-1]
            # print(self.)
            return self.ct_search()
```

```

import sys
file_name = sys.argv[1]

ex = sys.argv[2]
import pandas as pd
read_raw = read_data()
ct = snomed()
raw = read_raw.read_raw(ex)
count = 0
final_result = {}
for i in raw.keys():
    count += 1
    if count % size != rank:
        continue
    pre_data = raw[i]['processed']
    final_result[i] = ct.ct_string_process(pre_data)

final_result_to = comm.gather(final_result, root = 0)
writing_result = []
if rank == 0:
    datas = []
    for i in range(12):
        for all_keys in final_result_to[i].keys():
            tmp = {}
            tmp['left'] = all_keys
            tmp['right'] = final_result_to[i][all_keys]
            writing_result.append(tmp)

name = file_name+'.csv'
with open(name,'w',encoding='utf8', newline='') as f:
    w = csv.writer(f)

    w.writerow(writing_result[0].keys())
    for x in writing_result:
        w.writerow(x.values())

```

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or feedback?
BIS2.1	InsecureRequestWarning is being ignored in <code>ct_search</code> method, which could potentially be a security issue.	Instead of ignoring the warning, it's better to address the root cause of the issue, which is a problem with SSL verification. This can be done by properly configuring SSL verification in the system or by using a trusted SSL certificate.	Y	Putting the whole; system in Docker so that it does not need SSL. (Still working on it)
BIS2.2	<code>ct_string_process</code> method is updating the <code>self.name</code> variable, which is not necessary since it is only used in <code>ct_search</code> method.	Remove the line <code>self.name = string_for_ct[:-1]</code> and pass <code>string_for_ct[:-1]</code> directly as an argument to <code>ct_search</code> method. This will make the code cleaner and easier to understand.	Y	Yes
BIS2.3	The <code>count</code> the variable is being used to control the loop-in <code>ct_search</code> method, which is not necessary since the loop can be controlled directly using <code>pre_data</code> .	Remove the line <code>count = 0</code> and replace the loop for <code>ct_i</code> in <code>pre_data</code> : with for <code>idx, ct_i</code> in <code>enumerate(pre_data)</code> : and use <code>idx</code> to control the loop. This will make the code cleaner and easier to understand.	Y	Yes
BIS2.4	The <code>final_result_to</code> variable is a list of dictionaries, where each dictionary represents the final result of one process. This can cause issues when merging the results since the order of the dictionaries is not guaranteed.	Instead of creating a list of dictionaries, create a list of tuples, where each tuple represents a key-value pair. This will make it easier to merge the results later.	P	Not much operating time

(Solved in commit '[Update snomedct BIS 2.1-2.4](#)')

Login and Register Review:

Reviewer: Jiacheng Wang
 Developer: Mingjun Gao

[Login and Register](#)

```

@app.route("/login", methods=["POST"])
def login():
    data = request.get_json()
    username = data.get("username")
    password = data.get("password")
    cursor.execute('SELECT password FROM users WHERE username=%s', username)
    result = cursor.fetchone()

    if not result:
        # User not found
        return jsonify({'message': 'User Not Found'}), 403

    # Check password hash with bcrypt
    if bcrypt.checkpw(password.encode('utf-8'), result[0].encode('utf-8')):
        cursor.execute('SELECT id,username,firstname,lastname,email FROM users WHERE username=%s', username)
        result = cursor.fetchone()
        return jsonify({'message': 'Login successful', 'userinfo': {'userid': result[0], 'username': result[1], 'firstname': result[2], 'lastname': result[3], 'email': result[4]}}, 200
    else:
        # Passwords do not match
        return jsonify({'message': 'Invalid credentials'}), 403

@app.route("/register", methods=["POST"])
def register():
    data = request.get_json()
    username = data.get("username")
    password = data.get("password")
    email = data.get("email")
    cursor.execute('SELECT * FROM users WHERE username=%s OR email=%s', (username, email))
    result = cursor.fetchone()
    if result:
        return jsonify({'message': 'User or Email already exists'}), 403
    else:
        hashed = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt()).decode('utf-8')
        cursor.execute('INSERT INTO users(username, password, email, firstname, lastname) VALUES(%s, %s, %s, %s, %s)', (username, hashed, email, '', ''))
        con.commit()
        return jsonify({'message': 'User created successfully'}), 200

```

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or Feedback
BIS3.1	No error handling for missing or invalid request data in <code>login</code> and <code>register</code> functions.	Implement try-except blocks to handle exceptions raised by missing or invalid data. Return appropriate error messages to the user.	Y	Yes
BIS3.2	Plain text password is being sent over the network in <code>register</code> function.	Use HTTPS protocol to encrypt the network traffic, and please don't send plain text passwords over the network.	P	Sending the encrypted text
BIS3.3	SQL injection vulnerability in <code>register</code> function.	Use parameterized SQL queries or ORM to prevent SQL injection attacks.	P	Done in database
BIS3.4	The code is not modular and lacks separation of concerns.	Refactor the code to separate concerns, such as authentication logic and database access, into separate modules. Use appropriate design patterns (e.g., MVC or DAO) to structure the codebase.	Y	Y

(Solved in Sprint2Milestone2 commit 'add comment')

Mapping related Review

Reviewer: Jiacheng Wang

Mapping related functions

```
@app.route("/getmaps",methods=[ "GET"])
def getmap():
    userid = request.headers.get('Getmapping').split(' ')[1]
    cursor.execute("SELECT * FROM Mappings")
    result = cursor.fetchall()
    temp = []
    if result:
        for i in result:
            temp.append({'mapid':i[0],'userid':i[1],'username':i[2],'comment':i[3],'editdate':i[4].strftime('%Y-%m-%d %H:%M:%S')})
        return jsonify({'message': 'Map found','map':temp}), 200
    else:
        return jsonify({'message': 'Map not found'}), 403

@app.route("/deletemap/<string:mapping_id>/<user_id>",methods=[ "DELETE"])
def deleteMap(mapping_id,user_id):
    # return jsonify({'message': mapping_id}), 200
    cursor.execute("DELETE FROM Mappings WHERE id = %s", mapping_id)
    con.commit()
    file_path = os.path.join(app.config["PROCESS_FOLDER"], user_id,mapping_id+'.csv')
    os.remove(file_path)
    return jsonify({'message': 'Map deleted successfully'}), 200

@app.route("/getmapresult/<user_id>/<string:mapping_id>",methods=[ "GET"])
def getmapresult(user_id,mapping_id):
    path = os.path.join(app.config["PROCESS_FOLDER"], user_id, mapping_id+'.csv')
    with open(path, mode='r') as csv_file:
        csv_reader = csv.DictReader(csv_file)
        data = json.dumps([row for row in csv_reader])
    return data, 200

@app.route("/editmapping",methods=[ "POST"])
def editmapping():
    data = request.get_json()
    editinfo = data.get("editinfo")
    index = editinfo['index']
    userid = data.get("userid")
    mapid = data.get("mapid")
    path = os.path.join(app.config["PROCESS_FOLDER"], userid, mapid+'.csv')

    # Edit data
    newrow = [editinfo['raw'], editinfo['result'], editinfo['Flag']]
    # Edit the process csv file
    with open(path, mode='r', newline='') as f:
        reader = csv.reader(f)
        rows = list(reader)
        rows[index+1] = newrow
    with open(path, mode='w', newline='') as f:
        writer = csv.writer(f)
        writer.writerows(rows)

    with open('modify.csv', mode='r', newline='') as f:
        reader = csv.reader(f)
        rows = list(reader)
        found = False
        for row in rows:
            if newrow[0] in row:
                row[1] = newrow[1]
                found = True
                break
        if not found:
            rows.append(newrow[0:2])
    with open('modify.csv', mode='w', newline='') as file:
        writer = csv.writer(file)
```

```

writer.writerows(rows)
return jsonify({'message': 'Map edited successfully'}), 200

```

Issue ID	Issues	Suggestions	Solved (Y/N/P)	Tested
BIS4.1	No input validation in <code>getmap()</code>	Validate the <code>userid</code> before using it to fetch maps from the database. Return an error response if <code>userid</code> is not provided or not valid.	P	
BIS4.2	SQL statements should not be exposed directly in the code	Use prepared statements for SQL queries to prevent SQL injection attacks	N	
BIS4.3	Correct use of HTTP status codes	Instead of returning a 403 status code when no map is found, it would be more appropriate to return a 404 status code.	P	For future test
BIS4.4	More descriptive variable names	instead of using the variable 'temp', you could use 'mappings_list'.	Y	Yes

(Solved in Sprint2Milestone2 commit 'add comment')

ChatGPT-assisted Code Review

Notice: all of the code and fixed commits in branch 'sprint3_beta1'

Snomed CT connector Review:

Reviewer: Chatgpt
Developer: Lang Cao

snomed_ct.py

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or feedback?
BIS1.1	a. Naming: Variable names are generally descriptive, but there is a lack of clarity around the purpose of the <code>ct_search</code> and <code>ct_string_process</code> methods. b. Comment: There are no comments in the code to explain the purpose of the different methods or to provide context for the code.	a. Naming: Rename <code>ct_search</code> and <code>ct_string_process</code> to more clearly indicate their purpose. b. Comment: Add comments to explain the purpose of each method and provide context for the code.	Y	Fixed
BIS1.2	a. Bracket Usage: There are no bracket usage issues. b. Indentation: The indentation is consistent, but some lines are indented too far, making the code less readable. c. Long Line: Some lines are too long, which makes the code less readable.	a. Indentation: Adjust the indentation of some lines to make the code more readable. b. Long Line: Break up long lines to improve code readability.	Y	Fixed
BIS1.3	a. Dead Code: There is some commented-out code that is not used in the current version of the program. b. Duplication: There is no duplication in the code.	a. Dead Code: Remove the commented-out code that is not used in the current version of the program.	Y	Fixed
BIS1.4	a. Compute: There is no incorrect logic during system execution. b. Performance: It is unclear how efficient the <code>ct_search</code> method is, as it makes a request to an external server. This could be a potential performance bottleneck.	b. Performance: Consider caching the results of the <code>ct_search</code> method to avoid making repeated requests to the external server.	Y	Fixed

(Solved in commit '[Update BIS BIS2.1](#)')

read_ct Review:

Reviewer: Chatgpt
Developer: Lang Cao

<https://github.com/LeoBline/chatgptcodereview/pull/23>

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or feedback?
BIS2.1	1. Documentation Defects: a. Naming: The variable name "read" is not descriptive enough and could be improved. b. Comment: The code lacks any comments explaining what it does or how it works.	a. Naming: The variable name "read" can be changed to a more descriptive name such as "data_reader". b. Comment: Comments should be added to explain what the code does and how it works.	Y	Fixed

(Solved in commit [Update BIS BIS2.1](#))

read_data.py:

Reviewer: Chatgpt
Developer: Mingjun Gao

<https://github.com/LeoBline/chatgptcodereview/pull/23>

Issue ID	Issues	Suggestions	Solved (Y/N /P)?	Tested or Feedback

BIS3.1	<p>1. Documentation Defects:</p> <ul style="list-style-type: none"> a. Naming: The naming of the class and function names are not very descriptive. They could be improved to better reflect their purpose. b. Comment: There are some commented out code and print statements that could be removed to improve readability. Some of the comments are also not very descriptive and could be improved. 	<ul style="list-style-type: none"> • Improve the naming of the class and function names to better reflect their purpose. • Remove unnecessary commented out code and print statements. • Refactor the duplicated code within the <code>read_uil_list</code> function. • Consider adding more descriptive comments to improve readability and maintainability. 	Y	Fixed
BIS3.2	<p>1. Visual Representation Defects:</p> <ul style="list-style-type: none"> a. Bracket Usage: The bracket usage seems to be correct. b. Indentation: The indentation seems to be correct. c. Long Line: There are some long lines that could be split up to improve readability. 	<ul style="list-style-type: none"> • Split up long lines to improve readability. 	Y	Fixed
BIS3.3	<p>1. Structure Defects:</p> <ul style="list-style-type: none"> a. Dead Code: There is some commented out code that could be removed. b. Duplication: There is some duplication of code within the <code>read_uil_list</code> function that could be refactored. 	<ul style="list-style-type: none"> • Improve the comments to be more descriptive. 	Y	Fixed
BIS3.4	<p>1. a. Use Standard Method: There are no single-purpose code statements that would benefit from a standardized approach.</p>	no suggestion	P	Pass
BIS3.5	<p>1. Resource Defects:</p> <ul style="list-style-type: none"> a. Variable Initialization: The variable initialization seems to be correct. b. Memory Management: There is no explicit memory management in the code. 	no suggestion	P	Pass
BIS3.6	<p>1. Check Defects:</p> <ul style="list-style-type: none"> a. Check User Input: There is no user input in the code. 	no suggestion	P	Pass
BIS3.7	<p>1. Interface Defects:</p> <ul style="list-style-type: none"> a. Parameter: There are no incorrect or missing parameters when calling functions or libraries. 	no suggestion	P	Pass
BIS3.8	<p>1. Logic Defects:</p> <ul style="list-style-type: none"> a. Compute: There are no obvious incorrect logic during system execution. b. Performance: There is no obvious inefficiency in the algorithm used. 	no suggestion	P	Pass

(Solved in commit '[Update BIS3.1-3.8](#)')

Writing Review

Reviewer: Chatgpt

Developer: Mingjun Gao

<https://github.com/LeoBline/chatgptcodereview/pull/23>

Issue ID	Issues	Suggestions	Solved (Y/N/P)	Tested
BIS4.1	<p>1. Documentation Defects:</p> <ul style="list-style-type: none"> a. Naming: The class name "writing" is not descriptive enough and should be changed to something more meaningful. b. Comment: There are no comments in the code to explain the purpose of each method. 	<p>1. Documentation Defects:</p> <ul style="list-style-type: none"> a. Rename the class to something more meaningful, such as "DataWriter". b. Add comments to each method to explain its purpose and functionality. 	P	Pass
BIS4.2	<p>1. a. Dead Code: The commented out line in the "writing" method should be removed.</p> <p>b. Duplication: There are no duplicate code statements that can be refactored.</p>	<p>1. Structure Defects:</p> <ul style="list-style-type: none"> a. Remove the commented out line in the "writing" method. 	Y	Fixed

(Solved in commit '[Update BIS4.1-4.2](#)')

main Review

Reviewer: Chatgpt

Developer: Mingjun Gao

<https://github.com/LeoBline/chatgptcodereview/pull/28>

Issue ID	Issues	Suggestions	Solved (Y/N/P)	Tested
BIS4.1	a. Documentation Defects: a. Naming: The names of the variables and functions are descriptive and follow the PEP 8 naming conventions. b. Comment: The code has some comments, but they are limited and could be improved to better explain the purpose of the code.	<ul style="list-style-type: none">Add more comments to explain the purpose of the code and how it works.	Y	Fixed
BIS4.2	1. Visual Representation Defects: a. Bracket Usage: The code has correct bracket usage. b. Indentation: The code has consistent indentation. c. Long Line: The code has some long lines that could be split to improve readability.	<ul style="list-style-type: none">Split long lines of code to improve readability.	Y	Fixed
BIS4.3	1. Structure Defects: a. Dead Code: There is no dead code. b. Duplication: There is some duplication in the import statements for the 'json' module.	<ul style="list-style-type: none">Remove duplication in the import statements for the 'json' module.	Y	Fixed

(Solved in Sprint2Milestone2 commit '[Update main.py BIS4.1-4.3](#)')

Front-end review

Sprint 2 code Review

HomePage Review

Reviewer: Jiacheng Wang

Developer: Mingjun Gao

HomePage

```
<template>
  <el-container>
    <!-- Nav bar -->
    <el-header>
      <!-- Menu items -->
      <el-menu
        :default-active="activeIndex"
        mode="horizontal" @select="handleSelect">
        <el-menu-item index="1">Mapping System</el-menu-item>
        <el-menu-item index="2">Mappings</el-menu-item>
      <!-- Display user name in nav bar -->
      <div style="float:right; line-height: 60px; margin-right:20px;">
        <span style="margin-right:10px">{{ "Hi, "+userinfo.username }}</span>
      <!-- Dropdown for sign out and view user profile -->
      <el-dropdown>
        <i class="el-icon-user" style="margin-right: 15px"></i>
        <el-dropdown-menu slot="dropdown">
          <el-dropdown-item>Profile</el-dropdown-item>
          <el-dropdown-item><div @click="signout">Sign Out</div></el-dropdown-item>
        </el-dropdown-menu>
      </el-dropdown>
    </div>
    </el-menu>
  </el-header>
  <!-- Main content includes upload components-->
  <el-main class="main-content" style="margin-top: 20%;">
    <Upload></Upload>
  </el-main>
</el-container>
</template>

<script scoped>
import axios from 'axios';
// upload component
import Upload from './Upload.vue'

export default {
  name: "HomePage",
  components: {
    Upload
  },
  created () {
    this.checktoken();
  },
  data() {
    return {
      // menu active display index
      activeIndex: '1',
      // store user info after login success
      userinfo: {}
    };
  },
  methods: {
    // check token and login status if token is expired or not
    // if token is expired, redirect to login page
    checktoken(event){
      const tokenStr = localStorage.getItem('token');
    }
  }
}
```

```

localStorage.removeItem('file');
localStorage.setItem('file', '[]');
if (tokenStr !== null) {
  const tokenObj = JSON.parse(tokenStr);
  const userInfo = tokenObj.userinfo;
  const expireTime = tokenObj.expireTime;

  if (new Date().getTime() > expireTime) {
    localStorage.removeItem('tokenObj');
    this.$router.push("/login");
  }
  else this.userInfo = userInfo;
}
else{
  this.$router.push("/login");
}
},
// signout
signout(event){
  localStorage.removeItem('token');
  this.$router.push("/login");
},
// handle menu select
handleSelect(key, keyPath) {
  console.log(key, keyPath);
  if (key == 1) this.$router.push("/home");
  else if (key == 2) this.$router.push("/mapping");
}
}
};

</script>
<!-- scoped disable global css -->
<style scoped>
.main-content {
  display: flex;
  justify-content: center;
  align-items: center;
  position: relative;
  width: 100%;
  height: 100%;
}
</style>

```

Issue ID	Possible Issues	Suggestion	Solved(Y/N/P)?	Tested?
FIS1.1	Excessive use of inline styles	Suggest using a separate stylesheet to manage styles to enhance maintainability and modularity.	N	solve in sprint3
FIS1.2	The navigation bar selection style could use dynamic class binding	In <el-menu-item>, use the ':class' property of Vue.js to bind dynamic classes	P	
FIS1.3	Lack of route name consistency	Standardize route names to PascalCase or kebab-case to improve readability and consistency	Y	Yes

Github commit as "Comment front end codes" in sprint2_milestone2 Router review

Reviewer: Jiacheng Wang

Developer: Mingjun Gao

Index

```

import Vue from "vue";
import VueRouter from "vue-router";
import HomePage from "../components/HomePage.vue";
import Login from "../components/LoginPage.vue";
import Register from "../components/RegisterPage.vue";

```

```

import ViewMapping from "../components/ViewMapping.vue";
import MappingPage from "../components/MappingPage.vue";
Vue.use(VueRouter);
const routes = [
  {
    path: '/',
    name: 'index',
    component: HomePage
  },
  {
    path: '/home',
    name: 'HomePage',
    component: HomePage,
    meta: { requiresAuth: true }
  },
  {
    path: '/mapping',
    name: 'MappingPage',
    component: MappingPage,
    meta: { requiresAuth: true }
  },
  {
    path: '/login',
    name: 'Login',
    component: Login
  },
  {
    path: '/register',
    name: 'Register',
    component: Register
  },
  {
    path: '/viewmapping',
    name: 'ViewMapping',
    component: ViewMapping,
    meta: { requiresAuth: true }
  },
];
const router = new VueRouter({
  mode: "history",
  base: process.env.BASE_URL,
  routes,
});
// route guard to check if user is logged in
// if not logged in, redirect to login page
// if logged in, proceed to route
// https://router.vuejs.org/guide/advanced/navigation-guards.html#global-before-guards
router.beforeEach((to, from, next) => {
  const loggedIn = localStorage.getItem('token');
  if (to.matched.some(record => record.meta.requiresAuth) && !loggedIn) next('/login')
  else next()
});
export default router;

```

Issue ID	Possible Issues	Suggestion	Solved (Y/N /P)?	Tested?
FIS2.1	Lack of client configuration	Uncomment the eslint-disable comment and configure eslint to ensure consistent code style and readability.	P	
FIS2.2	Routes not in alphabetical order	Consider reordering routes in alphabetical order, or grouping them by their purpose or hierarchy, to improve readability.	N	solve in sprint3
FIS2.3	Unclear or generic route names	Use more descriptive and specific names for the routes, such as "DashboardPage" or "MappingDetailsPage", to improve readability and maintainability.	P	
FIS1.4	Login authentication using global navigation guards	In the router.beforeEach callback function, use store.getters.isLoggedIn to replace the locally stored validation	N	solve in sprint3

Github commit as "[Comment front end codes](#)" in [sprint2_milestone2](#)

Login page Review

Reviewer: Lang Cao
Developer: Mingjun Gao & Jiacheng Wang

Login Page

```
<template>
  <div class="login-container">
    <div class="login-box">
      <!-- login alert shows if login success or not -->
      <el-alert style="margin-bottom: 10px;" v-if="loginMessage" :title="loginMessage" :type="`${loginError ? 'error':'success'}`" show-icon></el-alert>
      <!-- login logo -->
      <a href="/home">
        
      </a>
      <!-- Login Form -->
      <h3 class="title">Login</h3>
      <el-form ref="loginForm" :model="loginForm" class="login-form">
        <el-form-item label="User Name" prop="username">
          <el-input v-model="loginForm.username" placeholder="Please Enter Username"></el-input>
        </el-form-item>
        <el-form-item label="Password" prop="password">
          <el-input type="password" v-model="loginForm.password" placeholder="Please Enter Password"></el-input>
        </el-form-item>
        <el-form-item>
          <div>
            <label style="vertical-align: middle;" :class="`checkbox ${remeberMe ? 'checked':''}`">
              <input type="checkbox" v-model="remeberMe">
            </label>
            <span style="vertical-align: middle;">Remeber Me</span>
          </div>
        </el-form-item>
        <el-form-item>
          <el-button class="login-button" @click="submitForm">Login</el-button>
        </el-form-item>
        <el-form-item>
          <p>Don't have an account?<router-link to="/register">Sign Up</router-link></p>
        </el-form-item>
      </el-form>
    </div>
  </div>
</template>

<script>
import axios from 'axios';
export default {
  name: "LoginPage",
  created () {
    this.getinfo();
  },
  data() {
    return {
      // login alert message
      loginMessage: "",
      // remeber me checkbox data
      remeberMe: false,
      // login error status
      loginError: false,
      // login form data username and password
      loginForm: {
        username: '',
        password: ''
      }
    };
  }
},
```

```

methods: {
    // login form submit function
    async submitForm (event) {
        try {
            const response = await axios.post('http://127.0.0.1:5000/login', {
                username: this.loginForm.username,
                password: this.loginForm.password
            })
            // remeber me function
            if (this.remeberMe) {
                localStorage.setItem('remeberuser', JSON.stringify(this.loginForm.username));
            }
            // update login status to success and show login success alert
            this.loginError = false
            this.loginMessage = response.data.message
            // set token to localstorage
            const expireTime = new Date().getTime() + 3600 * 1000*3;
            const tokenObj = {
                userinfo: response.data.userinfo,
                expireTime: expireTime
            };
            localStorage.setItem('token', JSON.stringify(tokenObj));
            console.log(JSON.parse(localStorage.getItem('token')).userinfo)
            // redirect to home page after 2s
            setTimeout(() => {
                this.$router.push("/home");
            }, 2000);
        } catch (error) {
            // update login status to error and show login alert
            this.loginError = true
            this.loginMessage= error.response.data.message
            // show login false alert for 2s
            setTimeout(() => {
                this.loginMessage = "";
            }, 2000);
        }
    },
    // remeber me function check localstorage if has remeber user
    // if has remeber user set remeber me checkbox to true
    // and set login form username to remeber user
    // if not set remeber me checkbox to false
    // and set login form username to empty
    getinfo(){
        const userInfo = localStorage.getItem('remeberuser');
        if (userInfo) {
            this.loginForm.username = JSON.parse(userInfo);
            this.remeberMe = true;
        }
    },
},
},
};

</script>
<!-- scoped disable global css -->
<style scoped>
/* login form css */
h3 {
    text-align: center;
    margin-bottom: 20px;
}

.uni-logo {
    display: block;
    width: 100px;
    height: auto;
    margin: 0 auto 20px;
}
.login-container {
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
}

```

```
}

.login-box {
  width: 400px;
  padding: 20px;
  background-color: #fff;
  border-radius: 5px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

.title {
  font-size: 24px;
  margin-bottom: 20px;
}

.login-form {
  width: 100%;
}

/* Modify Element UI default CSS */
.el-form-item__label {
  font-size: 14px;
}

.el-input__inner {
  height: 36px;
  font-size: 14px;
}

.el-button {
  width: 100%;
}

.el-button.login-button {
  background-color: #0d3978;
  border-color: #0d3978;
  color: #ffffff;
}
.el-button.login-button:hover {
  background-color: #061d3c;
  border-color: #061d3c;
  color: #ffffff;
}

/* checkbox */
.checkbox {
  display: inline-block;
  position: relative;
  width: 18px;
  height: 18px;
  background-color: #fff;
  border: 1px solid #dcdfec;
  border-radius: 2px;
  cursor: pointer;
  margin-right: 10px;
}
.checkbox.checked {
  background-color: #0d3978;
  border-color: #0d3978;
}
.checkbox.checked::after {
  content: "";
  position: absolute;
  left: 5px;
  top: 1px;
  width: 5px;
  height: 10px;
  border: solid #fff;
  border-width: 0 2px 2px 0;
  transform: rotate(45deg);
```

```

}
.checkbox:hover {
  border-color: #409eff;
}
.checkbox input[type="checkbox"] {
  display: none;
}
</style>

```

Issue ID	Possible Issues	Suggestion	Solved (Y/N /P)?	Tested?
FIS3.1	Duplicate data properties	It is not necessary to have both individual username and password properties and a loginForm object. Either remove the individual properties and use loginForm.username and loginForm.password in the template, or remove the loginForm object and use username and password in the template.	P	
FIS3.2	Misleading variable name	The loginError variable does not seem to represent an error state of the login form, but rather the success or failure of the login attempt. Consider renaming it to loginSuccess or something similar.	P	
FIS3.3	Missing validation	The form does not include any validation logic. Consider adding validation rules for the username and password fields, such as requiring them to be non-empty, or adding a "forgot password" link to allow users to reset their password.	N	solve in sprint3
FIS3.4	Missing catch block for axios request	The axios.post() method call should include a catch block to handle any errors that might occur during the request, such as network errors or server-side issues.	N	solve in sprint3
FIS3.5	Lack of separation of concerns	The getInfo() method and the rememberMe variables are not related to the login form logic and could be separated into a separate utility or Vuex module.	P	
FIS3.6	Using scoped attributes in CSS	Please consider moving global styles to a separate CSS file and then introducing them in Vue components to improve style maintainability and readability	P	

[Github commit as "Comment front end codes" in sprint2_milestone2](#)

Mapping page review

Reviewer: Jiacheng Wang
 Developer: Mingjun Gao

Mapping page

```

<template>
  <el-container>
    <!-- Nav bar -->
    <el-header>
      <el-menu :default-active="activeIndex" mode="horizontal" @select="handleSelect">
        <!-- Menu items -->
        <el-menu-item index="1">Mapping System</el-menu-item>
        <el-menu-item index="2">Mappings</el-menu-item>
        <!-- Display user name in nav bar -->
        <div style="float:right; line-height: 60px; margin-right:20px;">
          <span style="margin-right:10px">{{ "Hi, "+userinfo.username }}</span>
        <!-- Dropdown for sign out and view user profile -->
        <el-dropdown>
          <i class="el-icon-user" style="margin-right: 15px"></i>
          <el-dropdown-menu slot="dropdown">
            <el-dropdown-item>Profile</el-dropdown-item>
            <el-dropdown-item><div @click="signout">Sign Out</div></el-dropdown-item>
          </el-dropdown-menu>
        </el-dropdown>
      </el-menu>
    </el-header>
  </el-container>

```

```

        </div>
    </el-menu>
</el-header>
<el-main>
    <!-- table shows all users' mapping tasks -->
    <!-- set which column can be searched -->
    <el-table
        :data="mappings.filter(data => !search || data.mapid.toLowerCase().includes(search.toLowerCase()))"
        || data.editdate.toLowerCase().includes(search.toLowerCase()))"
        style="width: 100%"
        border
        stripe>
        <!-- bind with data -->
        <el-table-column align="center" prop="mapid" label="ID"></el-table-column>
        <el-table-column align="center" prop="username" label="Username" width="180"></el-table-column>
        <el-table-column align="center" prop="comment" label="Comment" width="180"></el-table-column>
        <el-table-column align="center" prop="editdate" label="Created At"></el-table-column>
        <el-table-column align="center" label="Action" width="360">
            <!-- search bar -->
            <template slot="header" slot-scope="scope">
                <el-input v-model="search" size="mini" placeholder="Type to search"/>
            </template>
            <!-- operation buttons for view mapping details, download mapping result and delete mapping
-->
            <template slot-scope="scope">
                <el-button type="primary" size="mini" @click="viewMapping(scope.row)">View</el-button>
                <el-button type="success" size="mini" @click="downloadMapping(scope.row)">Download</el-
button>
                <el-button type="danger" size="mini" @click="deleteMapping(scope.row.mapid)">Delete</el-
button>
            </template>
        </el-table-column>
    </el-table>
</el-main>
</el-container>
</template>

<script scoped>
import axios from 'axios';
export default {
  name: "MappingPage",
  created () {
    this.checktoken();
    this.getmappinginfo();
  },
  data() {
    return {
      // menu active display index
      activeIndex: '2',
      // user info
      userinfo: {},
      // mapping data
      mappings: [],
      // search data
      search:''
    };
  },
  methods: {
    // check token and login status if token is expired or not
    // if token is expired, redirect to login page
    checktoken(event){
      const tokenStr = localStorage.getItem('token');
      if (tokenStr !== null) {
        const tokenObj = JSON.parse(tokenStr);
        const userInfo = tokenObj.userInfo;
        const expireTime = tokenObj.expireTime;

        if (new Date().getTime() > expireTime) {
          localStorage.removeItem('tokenObj');
          this.$router.push("/login");
        }
      }
    }
  }
}

```

```

        else this userinfo = userInfo;
    }
    else{
        this.$router.push("/login");
    }
},
// signout
signout(event){
    localStorage.removeItem('token');
    this.$router.push("/login");
},
// get all mapping tasks
// if there is no mapping task, show empty table
// if there is mapping task, show all mapping tasks
// if there is error, show empty table
getmappinginfo(){
    const path = 'http://127.0.0.1:5000/getmaps';
    axios.get(path,{
        headers:{
            'Getmapping': 'Bearer ' + this userinfo['userid']
        }
    })
    .then(response => {
        this.mappings = response.data.map;
    })
    .catch(error => {
        this.mappings=[];
        console.log(error);
    });
},
// delete mapping task
// if delete successfully, refresh mapping task table
// if there is error, show error message
deleteMapping(id){
    const path = 'http://127.0.0.1:5000/deletemap/'+id+'/'+this userinfo['userid'];
    axios.delete(path)
    .then(response => {
        console.log(response.data.message);
        this.getmappinginfo();
    })
    .catch(error => {
        console.log(error);
    });
},
// view mapping details
// redirect to view mapping page
viewMapping(id){
    console.log(id);
    localStorage.setItem('mapid', id.mapid);
    localStorage.setItem('mapuserid', id.userid);
    this.$router.push("/viewmapping");
},
// download mapping result
// if download successfully, download csv file
// if there is error, show error message
downloadMapping(id)
{
    axios({
        method: 'post',
        url: 'http://127.0.0.1:5000/download',
        data: {
            file_id: id.mapid,
            userid: id.userid
        },
        responseType: 'blob'
    })
    .then(response => {
        const url = window.URL.createObjectURL(new Blob([response.data]));
        const link = document.createElement('a');
        link.href = url;
        link.setAttribute('download', id.mapid+'.csv');
    })
}

```

```

        document.body.appendChild(link)
        link.click()
    })
},
// handle menu select
handleSelect(key, keyPath) {
  console.log(key, keyPath);
  if (key == 1) this.$router.push("/home");
  else if (key == 2) this.$router.push("/mapping");
}
}
};
</script>

<style scoped>
.main-content {
display: flex;
justify-content: center;
align-items: center;
position: relative;
width: 100%;
height: 100%;
}
</style>

```

Issue ID	Possible Issues	Suggestion	Solved(Y/N/P)?	Tested?
FIS4.1	In-line styles	Please consider moving the inline styles to the <style> tag to improve maintainability and readability	P	
FIS4.2	Mixing JavaScript code and HTML	Could move form validation and processing logic to Vue.js methods to improve code readability and maintainability	P	
FIS4.3	Using hard-coded API endpoints	Consider using environment variables or a configuration file to store API endpoints for easier management	P	
FIS4.4	No error handling for API calls	Add proper error handling for API calls to handle possible failures and improve user experience	N	solve in sprint3
FIS4.5	No proper loading indication for fetching mapping data	Implement a loading spinner or a similar indication when fetching mapping data for better user experience	N	solve in sprint3

Github commit as "Comment front end codes" in sprint2_milestone2

Data visualization and table review

Reviewer: Jiacheng Wang

Developer: Mingjun Gao, Runzhe Hua and Jiacheng Wang

Visual and Getting Map result

```

<script>
import axios from 'axios';
import * as echarts from 'echarts';
var temp = 0;
export default{
  name: "ViewMapping",
  created(){
    this.checktoken();
    this.getmapresult();
  },
  data(){

```

```

        return{
            userinfo: {},
            search: '',
            tableData: [],
            chartData: [
                { value: 0, name: 'no match' },
                { value: 0, name: 'match' },
            ],
            chartData1: [
                { value: 0, name: 'UIL' },
                { value: 0, name: 'SNOMED CT' },
            ],
            // dialog
            editDialogVisible: false,
            deleteDialogVisible: false,
            form: {
                raw: '123',
                result: '',
                Flag: '',
            },
            formLabelWidth: '120px'
            // mapresult: []
        };
    },
    methods: {
        checktoken(event){
            const tokenStr = localStorage.getItem('token');
            if (tokenStr !== null) {
                const tokenObj = JSON.parse(tokenStr);
                const userInfo = tokenObj.userinfo;
                const expireTime = tokenObj.expireTime;

                if (new Date().getTime() > expireTime) {
                    localStorage.removeItem('tokenObj');
                    this.$router.push("/login");
                }
                else this.userInfo = userInfo;
            }
            else{
                this.$router.push("/login");
            }
        },
        // signout
        signout(event){
            localStorage.removeItem('token');
            this.$router.push("/login");
        },
        // get mapping details and set charts data
        getmapresult(){
            const tokenStr = localStorage.getItem('token');
            const tokenObj = JSON.parse(tokenStr);
            const path = 'http://127.0.0.1:5000/getmapresult/' +localStorage.getItem('mapuserid') + ' +' +localStorage.getItem('mapid');
            axios.get(path)
            .then(response => {
                this.tableData = response.data;
                this.chartData[0].value = 0;
                this.chartData[1].value = 0;
                this.chartData1[0].value = 0;
                this.chartData1[1].value = 0;
                for (var i = 0; i < this.tableData.length; i++){
                    this.tableData[i].index = i;
                    if (this.tableData[i].result == 'Non-Match')
                    {
                        this.chartData[0].value += 1;
                    }
                    else{
                        this.chartData[1].value += 1;
                    }
                    if(this.tableData[i].Flag == 'UIL')
                    {
                
```

```

        this.chartData1[0].value += 1;
    }
    else{
        this.chartData1[1].value += 1;
    }
}
const pieChart = echarts.init(this.$refs.pieChart);
const sourcepieChart = echarts.init(this.$refs.sourcepieChart);
const options = {
    title: {
        text: 'Mapping Result',
        left: 'center',
    },
    tooltip: {
        trigger: 'item',
        formatter: '{a} <br/>{b} : {c} ({d}%)',
    },
    legend: {
        orient: 'vertical',
        left: 'left',
        data: ['match', 'no match'],
    },
    series: [
    {
        name: 'Mapping Result',
        type: 'pie',
        radius: '55%',
        center: ['50%', '60%'],
        data: this.chartData,
        emphasis: {
            itemStyle: {
                shadowBlur: 10,
                shadowOffsetX: 0,
                shadowColor: 'rgba(0, 0, 0, 0.5)',
            },
        },
    },
    ],
},
];
const sourceoptions = {
    title: {
        text: 'Mapping Source',
        left: 'center',
    },
    tooltip: {
        trigger: 'item',
        formatter: '{a} <br/>{b} : {c} ({d}%)',
    },
    legend: {
        orient: 'vertical',
        left: 'left',
        data: ['UIL', 'SNOMED CT'],
    },
    series: [
    {
        name: 'Mapping Source',
        type: 'pie',
        radius: '55%',
        center: ['50%', '60%'],
        data: this.chartData1,
        emphasis: {
            itemStyle: {
                shadowBlur: 10,
                shadowOffsetX: 0,
                shadowColor: 'rgba(0, 0, 0, 0.5)',
            },
        },
    },
    ],
},
],
color:[ '#68b8d9', '#f9c14e' ]
};

```

```

        sourcepieChart.setOption(sourceoptions);
        pieChart.setOption(options);
    })
    .catch(error => {
        console.log(error);
    });
},
// click edit button to update edit form
handleEdit(index, row) {
    this.form = row;
    console.log(index, row);
},
// click delete button to update delete form
handleDelete(index, row) {
    this.form = row;
    console.log(index, row);
},
// set filter tag for table
filterTag(value, row) {
    return row.Flag === value;
},
// set index for table
indexMethod(index) {
    return index+1;
},
// dialog for cancel edit
handleClose(done) {
    this.$confirm('Are you sure to cancel the edit?')
    .then(_ => {
        this.cancelEdit();
        done();
    })
    .catch(_ => {});
},
// update mapping details data if cancel edit
cancelEdit(event){
    this.getmapresult();
},
// update mapping details data if edit success
editmapping(event){
    const path = 'http://127.0.0.1:5000/editmapping';
    axios.post(path, {
        editinfo: this.form,
        userid: this.userinfo["userid"],
        mapid: localStorage.getItem('mapid')
    })
    .then(response => {
        console.log(response.data.message);
        this.getmapresult();
    })
    .catch(error => {
        console.log(error);
    });
},
// update mapping details data if delete success
deletemapping(event){
    const path = 'http://127.0.0.1:5000/deletemapping';
    axios.post(path, {
        mapinfo: this.form,
        userid: this.userinfo["userid"],
        mapid: localStorage.getItem('mapid')
    })
    .then(response => {
        console.log(response.data.message);
        this.getmapresult();
    })
    .catch(error => {
        console.log(error);
    });
},

```

```

        // handle menu select
        handleSelect(key, keyPath) {
            console.log(key, keyPath);
            if (key == 1) this.$router.push("/home");
            else if (key == 2) this.$router.push("/mapping");
        }
    }
}

</script>

```

Issue ID	Issue	Suggestion	Solved(Y/N/P)?	Tested?
FIS5.1	In-line styles	Consider moving the inline styles to the <style> tag to improve maintainability and readability	P	
FIS5.2	Directly modifying the DOM	Please avoid directly modifying the DOM; instead, use Vue.js methods and data-binding for a better coding practice	P	
FIS5.3	Using var instead of let or const	Replace var with let or const to follow ES6 best practices and avoid potential issues with variable scoping could be better	P	
FIS5.4	Using localStorage for storing sensitive data	Consider using a more secure method for storing sensitive data, such as Vuex or encrypted cookies	N	solve in sprint3

Github commit as "Comment front end codes" in sprint2_milestone2

Testing & Evaluation

[Test Case](#)

Test Case

Front-end and Back-end Tests

Test Case ID	Test Page	Test Case Description	Test Steps	Test Data	Precondition	Results	Pass /Fail	Tester
FBTC001	Register Page	Verify the register function.	<p>Positive test</p> <p>1. go to the login page. 2. click the "Sign up" Button. 3. input the Username, Email, and Password, and Confirm the Password. 4. click the "Register" button. 5. go back to the login page. 6. login with the new account.</p> <p>Negative Test</p> <p>The same</p>	<p>Positive test</p> <p>Username =admin1 Email= yake@student.unimelb.edu.au Password = 123456 Confirm Password = 123456</p> <p>Negative test 1</p> <p>Username =admin1 Email= yake@student.unimelb.edu.au Password = 123456 Confirm Password = 12345</p> <p>Negative test 2</p> <p>Username =admin1 Email= yestudent.unimelb.edu.au Password = 123456 Confirm Password = 12345</p>	FBTC002	<p>Positive</p> <p>It returns to the login page and I can log in with this new account.</p> <p>Negative</p> <p>Register to Fail</p>	Pass all	Yanxi Ke
FBTC002	Login page	Verify Login	<p>Positive test</p> <p>1. go to the login page. 2. input the User Name 3. input the password. 4. click the login button.</p> <p>Negative test</p> <p>The same</p>	<p>Positive test</p> <p>Username =admin1 Password = 123456</p> <p>Negative test 1</p> <p>Username =admin Password = 123456</p> <p>Negative test 2</p> <p>Username =admin1</p>		<p>Positive</p> <p>It successfully logs in and goes to the upload page.</p> <p>Negative</p> <p>Login fails</p>	Pass all	Yanxi Ke

			Password = 12345					
FBTC003		Verify log out	1. click the upper right user name button 2. click the "logout" button		FBTC002	it success logs out and goes back to the log-in page.	Pass	Yanxi Ke
FBTC004	Upload page	Verify the uploaded TXT file.	Positive test 1. click the upload button 2. select the file 3. click to upload Negative test 1. the same	Positive test upload txt file Negative test upload CSV file	FBTC002	Positive It should accept the file and change its status to processing. Negative Display error	Pass all	Lang Cao
FBTC004		Verify the upload of other files.	1. click the upload button 2. select the file		FBTC002	It rejects invalid file formats and provides an error message	Pass	Lang Cao
FBTC005		Verify the upload of many files	1. click the upload button 2. select the files		FBTC002	It can store all file ready for processing	Pass	Lang Cao
FBTC006		Verify the mapping result	1. Click the "Mappings" button. 2. Click the View button corresponding to the uploaded file id line.		FBTC002, FBTC009	it should show the visualization page and the details of the result.	Pass	Lang Cao
TC007		Modify a mapping result	1. Click Raw Text = "LRTI"s Edit Button. 2. input the new data in modified data = "respiratory tract" 3. Click the save button.	Raw Text = "LRTI" modify data = "respiratory tract"	FBTC002, FBTC006, FBTC009	it shows the new data in the corresponding raw data's target text.	Pass	Jiacheng Wang
TC008		Verify the export of the mapping result	1. Click the "Mappings" button. 2. Click the Download button corresponding to the uploaded file id line.		FBTC002, FBTC004, FBTC009	the computer downloads the finished files.	Pass	Jiacheng Wang
FBTC009		Verify mapping history (dashboard)	1. Click the "Mappings" button.		FBTC002	it shows the mapping page. and all the upload process history	Pass	Jiacheng Wang
FBTC010		Test system performance	1. Click the "Mappings" button. 2. Click the View button corresponding to the uploaded file id line.		FBTC002, FBTC009	The system should be able to handle multiple users uploading and modifying files simultaneously without crashing or significant slowdowns	Pass	Runzhe Hua
FBTC011		Verify function to show the confidence (distance) of the result from other possible results	1. Click the View button			The mapping result should also indicate the distance between the mapping result and the raw text and the distances between the raw text and other possible descriptions in SNOMED-CT.	Fail	Runzhe Hua
FBTC012		Verify the mapping result's visualization	1. Click the "Mappings" button. 2. Click the View button		FBTC002, FBTC009	it should show the visualization page about the accuracy and Specific matching category pie chart.	Pass	Lang Cao

			corresponding to the uploaded file id line.				
FBTC0013		Verify the legitimacy detection function for the modified value	1. Click the Modify button 2. type text			It should only be able to accept phrases that exist in SNOMED CT. If not one of the phrases should be prompted to enter legal content.	Fail Lang Cao

Note: FBTC0013 will be fixed in the next sprint

Mapping Algorithm Tests

Testing ID	Input	Expecting output	Actual Output	Pass or Fail	Description	Tester
MATC001	Sinusitis	Chronic sinusitis	Chronic sinusitis	Pass	SNOMED CT return sinusitis. In UIL, there are <ul style="list-style-type: none"> • Chronic sinusitis • Invasive fungal sinusitis • Uncomplicated acute rhinosinusitis • Complicated acute rhinosinusitis The result should be Chronic sinusitis because it has the least edit distance	Lang Cao
MATC002	Constipation	Constipation	Non-Match	Fail	It can be searched by SNOMED CT. But it does not mention in the UIL. This is a medical word.	Lang Cao
MATC003	Check up	Non-Match	Non-Match	Pass	This is a Non-medical word so it is Non-Match.	Lang Cao

Note: MATC002 will be fixed in the next sprint

Medical Data Classifier Tests

Test Case ID	Test Description	Test Steps	Test Data	Precondition	Results	Pass /Fail	Tester
MDTC 001	Test legitimate medical text and non-related text	1. Import the model from medical data class 2. Create an instance of the classifier and pass the test data 3. See whether the result follows the exception	medical text {'cardiology', 'hypertension', 'diabetes'} non-related {'apple', 'banana', 'grape'}	None	All the medical text and non-related pass the test. As all medical text got "True" as a return, and all non-related text got "False".	Pass	Jiacheng Wang
MDTC 002	Test whether the evaluation function returns the correct result.	1. Import the model from medical data classifier 2. Create a instance from the class and using the internal function evaluate_model(clf, vectorizer, X_test, y_test) 3. Calculate and check manually using a calculator(With 100 instances)	Prediction result{ True, False, True, False, True, True,... } Groudtruth{ True, True, True, False, True, True,... }	MDTC 001	The results of the program match the results of the manual calculations using the calculator	Pass	Jiacheng Wang

Code review checklist

This table provides a comprehensive code review checklist, covering various aspects of code quality, such as maintainability, error handling, security, thread safeness, resource leaks, control structures, and reusability.

Category	Checklist
Maintainability	<ul style="list-style-type: none">• Is the code understandable?• Ensure sufficient comments for code readability.• Is the code consistent with Python conventions?• Are functions independent of each other?
Error Handling	<ul style="list-style-type: none">• Is the code consistent with Exception Handling Conventions?• Are exceptions properly handled in the call stack to prevent stack traces from being displayed to the user?• Is exception handling employed throughout the code?
Security	<ul style="list-style-type: none">• Utilize secure connections like HTTPS over HTTP when possible.• Ensure that passwords are not stored in the code.• Are there any security issues within the code?
Thread Safeness	<ul style="list-style-type: none">• Is the code thread-safe?• Are deadlocks prevented in the code?
Resource Leaks	<ul style="list-style-type: none">• Are resources properly released in the code?• Ensure resources are not released multiple times.• Are efficient classes utilized for resource handling?
Control Structures	<ul style="list-style-type: none">• Are infinite loops avoided in the code?• Are loops designed to iterate the correct number of times?
Reusability	<ul style="list-style-type: none">• Are existing libraries leveraged effectively?• Ensure proper initialization of variables and objects.• Eliminate unreferenced or unused variables and objects.• Is the code sufficiently abstracted and generalized?• Can the code be considered for reusability?• Encapsulate reusable code in functions.

Coding Standards

1. JavaScript coding Standards
 - a. [JavaScript standards](#)
2. HTML coding Standards
 - a. [HTML Coding Standards](#)
3. CSS coding standards
 - a. [CSS Coding Standards](#)
4. Python coding standards
 - a. [Python coding Standards](#)
5. General Front-end standards
 - a. [Front-end Standards](#)
6. Back-end guide
 - a. [Backend-end Standards](#)
7. GitHub Guide
 - a. [Github Guide](#)

Risk Management

Risk Category	Risk Description	Impact	Likelihood	Risk Mitigation Strategy
Technical Risk	Input data quality is low	Medium	Medium	Perform data cleaning and validation before mapping.
Technical Risk	Mapping result is not accurate	High	Low	Use better algorithms. And users can modify incorrect mappings to improve the system algorithm.
Technical Risk	System Failure	High	Low	The system should be designed with fault tolerance.
Project Management Risk	Clients dissatisfied with deliverables	High	Low	Regularly demonstrate the deliverables to clients during the project to obtain their feedback and make adjustments and improvements in time.
Project Management Risk	Changes in project requirements	Medium	Medium	Regularly meet with stakeholders to discuss project progress and any changes in requirements.

Product

This page shows the final product video.

Notice: The whole product is put in a container by Docker. Therefore, make sure the whole process operates Docker Desktop.

In readme file

1. 'Setup for delivery' is the explanation for the setup of everything
 2. 'TIPS' includes some future actions
 3. 'development only' means this is for development setup (Not for delivery).
- This is the final product video.
 - This is the final product of our project.
 - Run the container called 'src' and then type 'localhost:3000' in the browser or click '3000'.
 - Product demo Youtube Link: <https://youtu.be/7EDKnoxz-is>
 - Product Video Youtube Link: <https://www.youtube.com/watch?v=ZZbfPFLMKIE>
 -
 - **Product setup video:**
 - Users only need to click setup.bat so that it will install automatically.
 - After installing everything successfully, it will open the login page.
 - Please check everything in container 'src' is running.
 - System requirement: at least 8GB Memory.
 - If there is insufficient memory, it will be stopped installing or installing fails.
 - Youtube Link: <http://youtube.com/watch?v=Cg8jEl47OQg>
 -
 - **Presentation Video**
 - This is the final presentation video including some feedback.
 - Youtube Link: <https://youtu.be/5Z6YU8VZ2vY>
 -

Presentations

Digital Health

COMP90082
Software Project

Lang Cao Mapping System Developer
Yanxi KE Developer
Mingjun GAO Developer
Jiacheng WANG Testing Developer
Runzhe Hua Developer



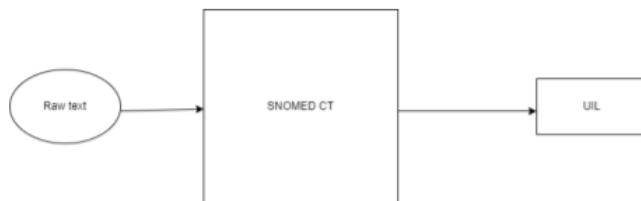
Overview

- Issue: Diverse formats of medical notes impede data standardization
- Solution: System transforming medical notes into a unified, official format



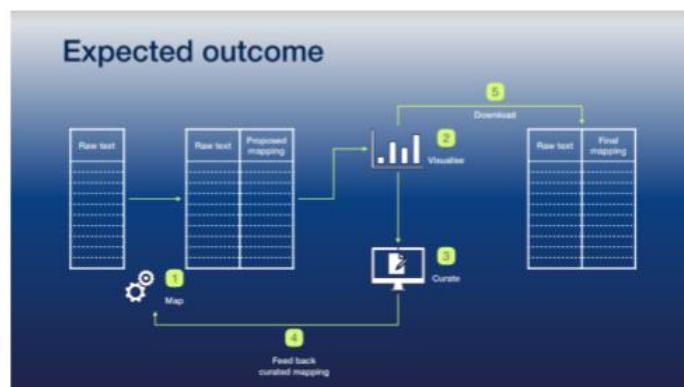
Our Goal and Main Process

- Goal: Map all relevant raw medical data to the Universal Indications List (UIL)
- Process
 - Using the Capabilities of SNOMED CT to Collect Potential Terms
 - A two-step verification process using UIL, ensuring the accuracy and relevancy of mapped terms
- Example result:
 - “skin, Infection”, “Cutaneous, bacterial invasion”, “Dermal, infection detected”-----> “infection of skin”



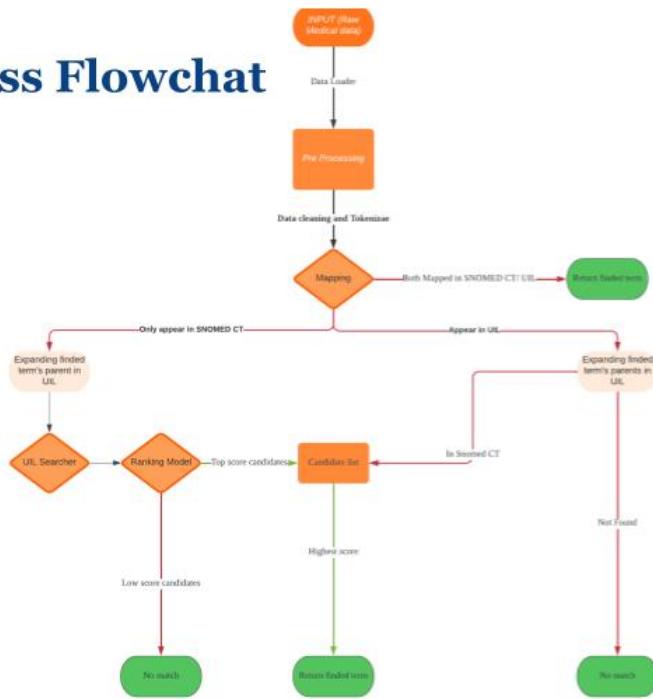
Key functions and Benefits

- Key functions: Data intake, standardization, organization, mapping, result output
- Outcome: Varied data transformed into a unified format

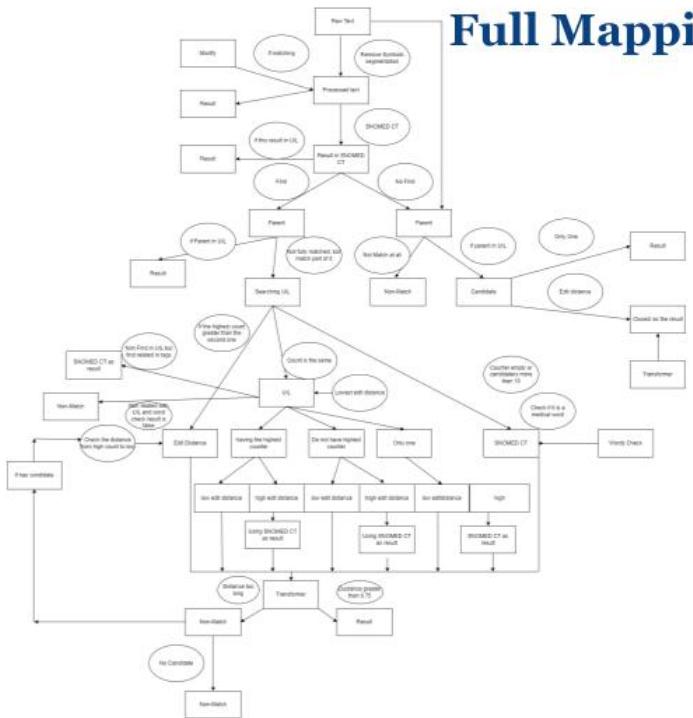




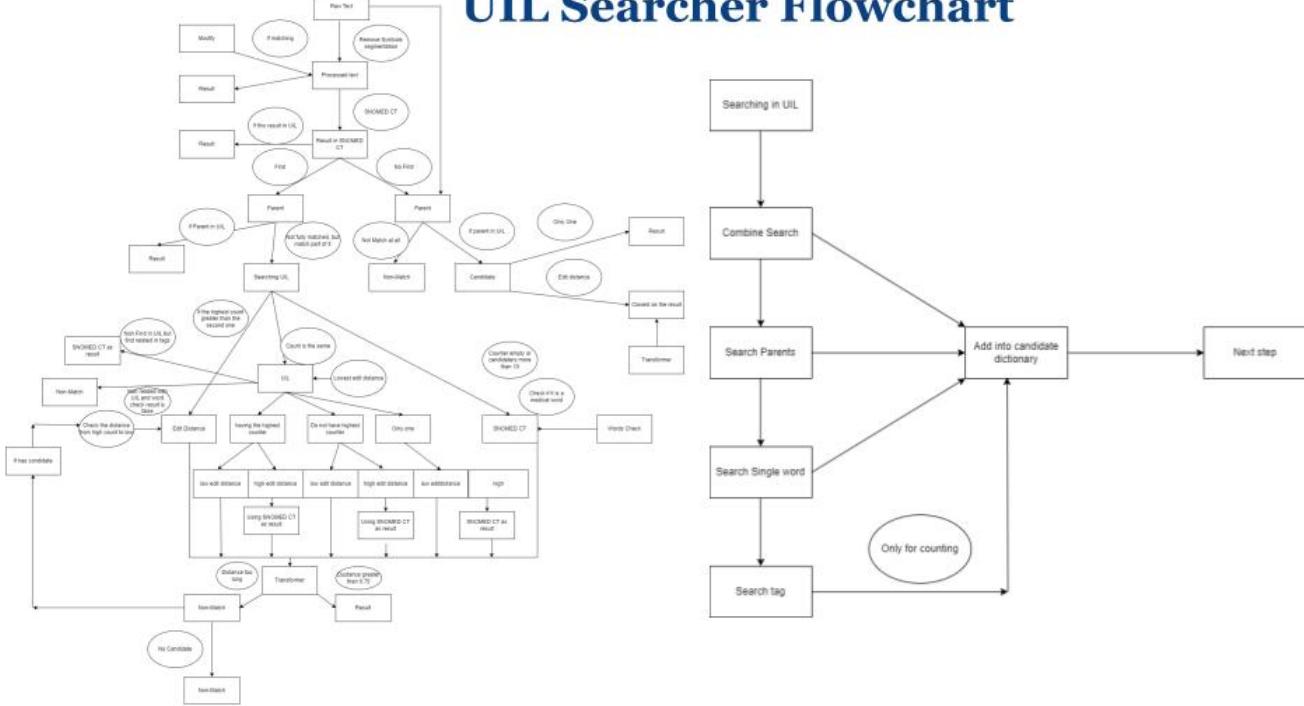
Mapping Process Flowchart



Full Mapping Process Flowchart



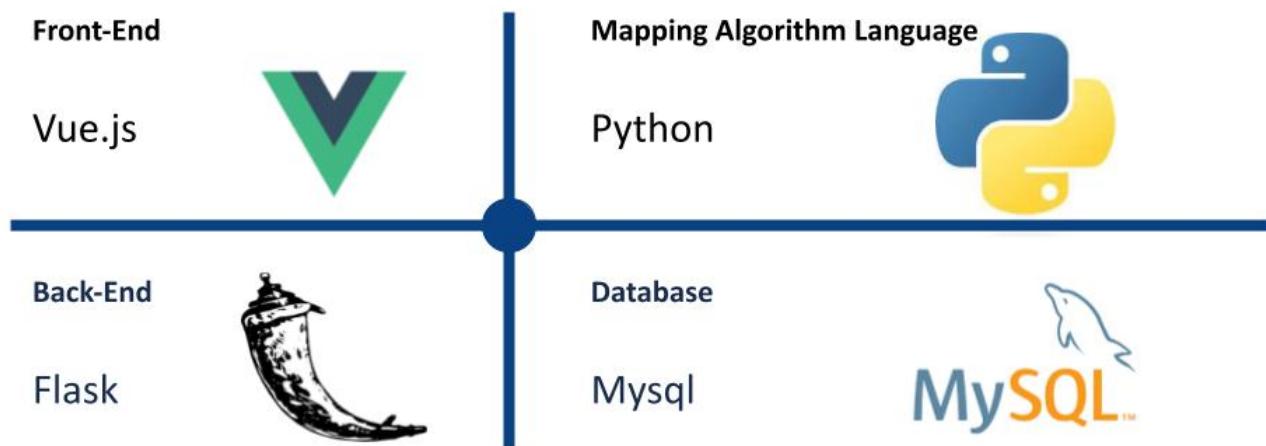
UIL Searcher Flowchart



Technology Use



Technology Use GENERAL



Technology Used in Mapping Algorithm

- SNOMED-CT
- Scispacy -en_ner_bionlp13cv_md F1=76
- Transformer -ClinicalBERT Base on 1.2B medical words F1=82
- MPI4PY
- Edit distance

Challenge



SNOMED CT Installation

An error in the configuration file on the Ontoserver official website

```
version: '3'
volumes:
  onto:
    driver: local
  pgdata:
    driver: local
services:
  db:
    image: postgres:12
    volumes:
      - pgdata:/var/lib/postgresql/data
    healthcheck:
      test: ["CMD-SHELL", "pg_isready -U postgres"]
      interval: 10s
      timeout: 5s
      retries: 5
    environment:
      - POSTGRES_HOST_AUTH_METHOD=trust
  ontoserver:
    image: quay.io/semrc/ontoserver:ctza-6
    container_name: ontoserver
    read_only: true
    security_opt:
      - no-new-privileges
    depends_on:
      - db
    ports:
      - "8443:8443"
      - "8080:8080"
    environment:
      # These two lines are specific to deployment in Australia only
      - authentication.oauth.endpoint.client_id=0-NCTS_CLIENT_ID
      - authentication.oauth.endpoint.client_secret=0-NCTS_CLIENT_SECRET
    #     - JAVA_OPTS=-Xmx2G           # Minimum
    #     - JAVA_OPTS=-Xmx8G          # Preferred
  volumes:
    - onto:/var/onto
    - /tmp
    - /var/log
```

Data Mapping with UIL

Raw text: Constipation

Candidate list: None

SNOMED CT result: Constipation

Final Result: Constipation

Reason: This is a medical word, so the result should be the return of the SNOMED CT because it is not mentioned in UIL

Deployment on Windows Server via Docker

Docker images can only be installed in Windows Subsystem for Linux (WSL).

By default, WSL installs in a specific folder on the system drive.

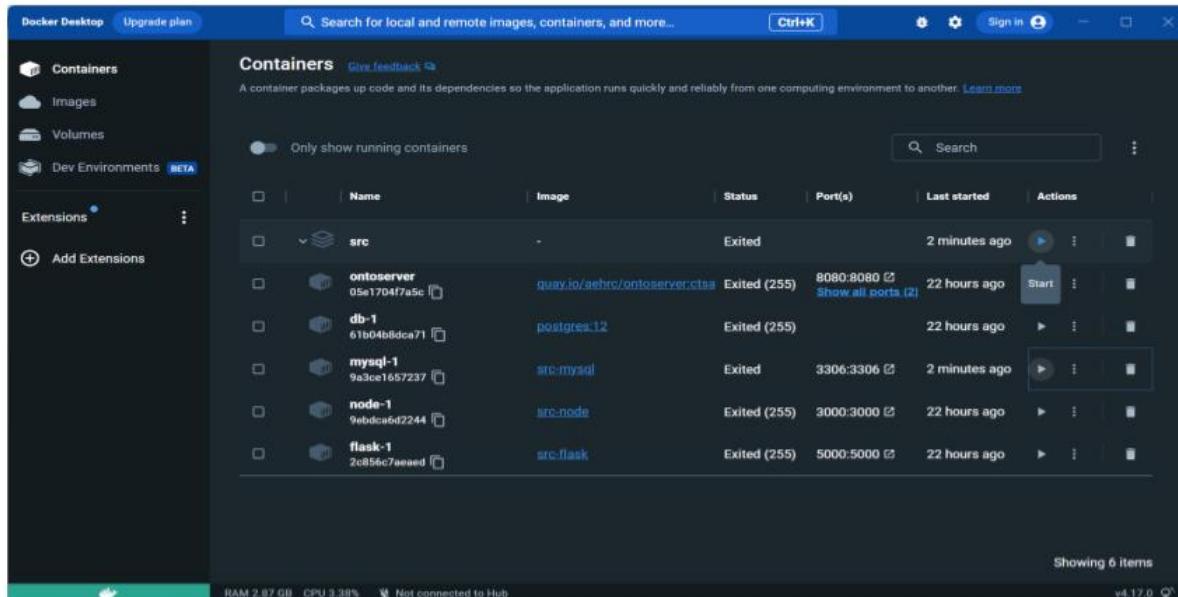
Deployment



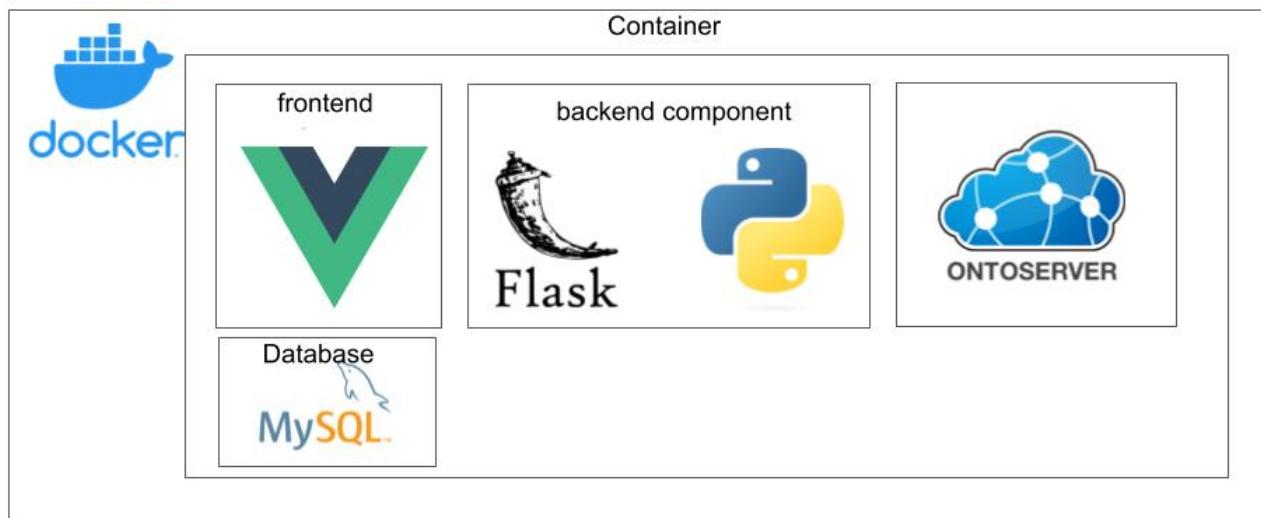
Why Docker?

Deploy on Docker	Deploy on Windows System
Download and Install docker	Download and Install docker
Install WSL	Install WSL
docker compose up -d	Install python and requirement packages
	Install NodeJS for windows
	docker compose up -d
	Installing Mysql

Running



Deployment Structure



Thank you for your attention



Research

- [Research items](#)

Research items

Name	Description	Fit the project? (Y/N/U)	Feedback
Transient Global Amnesia (TGA)	Medical vocabulary	U	Data quantity and quality: Not clear do we have enough good quality and sufficient quantity of TGA Data balance: Given the rarity of TGA, it could be hard to balance the weight between other diseases and TGA.
Transformer	RNN model	Y	Data processing speed: RNN could be slow when processing medical text-based data but with future optimization, and relatively simple medical notes, the final speed should be acceptable. Potential: RNN demonstrated excellent results in fields of NLP. It can handle sequence data, and it's quite robust to noise and variations in the data, making it a promising technology for future development. Limitations due to insufficient data: RNN requires a large amount of data to train the model, but the data from the client is too less to train the model.
Distance-based algorithm for mapping medical notes	Edit distance	N	After trying out several distance metrics, including Edit Distance, Euclidean Distance, Cosine Similarity, and Tree Edit Distance, we found that none of them performed well in converting raw medical data into a professional vocabulary format. It appears that these methods may not be suitable for the complexity and specificity of medical terminology. Further research is needed to explore alternative techniques for this task.
	Euclidean Distance		But the client asked us to use edit distance as visualisation. It works not very well in mapping algorithms but this is a good method for visualization.
	Cosine Similarity		
Medical database for classifier (NIH, CDC, WHO, NLM)	Tree Edit Distance		
	Need to classify whether the input RAW text is medical data	Y	The WHO's International Classification of Diseases (ICD) data was ultimately selected as the medical dataset for the text classifier due to several key advantages. Firstly, the ICD data is freely accessible, which allows researchers and developers to easily obtain and utilize the information without any financial burden. And the dataset is comprehensive, covering a wide variety of diseases and conditions, making it ideal for developing a versatile classifier. The ICD database also is consistently updated by the WHO, ensuring that the information remains current and relevant, which contributes to the ongoing accuracy and effectiveness of the classifier.

Y - Yes

N - No

U - Unclear

Project Details

Analysis of clinical documentation is critical for many digital health projects but extracting information from free-text clinical notes can be difficult. Our team is developing a platform that will assess if a medication has been appropriately prescribed to a patient.

Typically, the rationale for prescribing a particular medication – the reason for prescription – is commonly provided by clinical practitioners in short free-text strings. We are interested in normalising those short strings by mapping them onto a knowledge base of canonical clinical terms, known as SNOMED CT (<https://www.snomed.org>). For this, we intend to use a mapping tool called Ontoserver (<https://ontoserver.csiro.au/site/>) developed by CSIRO which can be deployed locally and tuned to specific data.

We are looking for a team of software engineers to set up and customise the tool and design a platform that will allow a human-in-the-loop to manually review the results of the mapping, make corrections, and feed these back to re-train the system. We are keen on exploring whether the tool can be deployed using Docker inside an existing virtual machine so we can turn it into a reusable component of our text analytics pipeline.

The project would suit students that are interested in text analytics and curious to get exposed to real-world healthcare data and creating scientific software.

Industry Partner: Dr Daniel Capurro