Hirushi Rajapakse
Birkbeck University 2022

Biocomputing II – Group 6

Approach to the project

The group project was set out to be conducted in highly a collaborative manner, where it required comprehensive data extraction, processing, querying and presenting in a web page. Group 6 instantaneously recognised this project requires efficient connection and merging between each layer to successfully produce a web page with all necessary requirements. We all agreed to have a dedicated lead for each layer and to work collectively so we all can benefit from each other's expertise, knowledge and understand what has been set up at each layer.

Initial approach to this project involved fully understanding the requirements of this project. Our first goal was to create a Github repository and successfully connect. The group first spent time matching all web page requirements against data present in the GenBank file for chromosome 1. We all agreed to make a start on business layer together as this was the foundation for all other application layers.

Interaction with the team

Initially group 6 interacted mostly during lectures to discuss the requirements, task allocation and preparation of dummy data in the form required by the API. During the first lecture all group members successfully managed to configure GitHub and create a Git repository. In Git hub we created active branches for each team members Amber, Denzel, Hirushi and Farah and the default branch was set to main.

During the second lecture we thoroughly investigated the project requirements and allocated each team member with their designated layer. All four team members collaborated efficiently, effectively and had great team dynamics. By the end of evening the group managed to prepare dummy code for API in data access tier and business logic tier that was presented during the lecture following week. The group prepared a PowerPoint presentation to present the dummy API and a simple example of how the final web page might look like. The group was very content as Dr. Andrew Martin recognised that we were amongst the very few groups that managed to fully understand each entry and prepare and present that in the dummy code for business logic and database layer.

The team also documented all the steps from connecting to Git hub, web page requirements and team reflections on google docs so that all team members can access content, add content and track progress. This document is attached part of this submission as appendix 2. In the upcoming weeks we all worked between each layer in the same collaborative manner where the lead developer chairing the session. Each group member had different levels of programming knowledge and working collaboratively improved my knowledge in coding, HTML, CSS and made it a great educational session and an enjoyable working atmosphere.

The group also met at Birkbeck university library as we discovered working remotely in evenings posed difficulties in finishing a task that was set out. We agreed to meet in the morning and worked through to the end of the day finishing business logic and database layer. Finally, we met and worked for a fifteen-hour day at Birkbeck library to connect each layer and complete front end. During this process I prepared a team log, and this is attached as appendix 1.

Overall project requirements

Database logic layer required accessing MySQL database to store and work from the GenBank file that had very large compatibility. We focused on extracting all gene identifiers such as gene, protein product name and chromosomal location for each GenBank accession number. At this stage we also discussed how it will appear at the front end in three different stages of search page, list all page and a detailed a page. This process allowed us to fully understand what this project required, how to transform the data in database and business logic layer and visualise at the front end.

Requirements for my contribution

I was responsible for producing a web-based interphase that will query the business logic layer to produce the front page, list all page and the detailed page. The CGI scripts for listall.cgi and search.cgi completed the process of extracting each entry from business logic layer to present as multiple detailed tables in a HTML format. I put together the index.py file which consisted of adding images, displaying all entries and search feature for database to find an GenBank accession for individual entries such as gene identifier, protein product names, GenBank accession, or chromosomal location. My work also involved presenting the functions for restriction enzymes sites, percentage of codon counts and amino acids sequences in a table format. As mentioned above I also collectively contributed towards completing the database and business logic layers.

Performance of the development cycle

First, we fully investigated the project requirements and the GenBank file for each entry and made relationships amongst them by identifying what are the primary keys, types of data required at each layer and how to extract them. Appendix 1 shows how we made progress over the weeks by moving into the next layer and between each layer. We anticipated each layer will take 2 weeks to complete, however we soon realised it took lot longer specially for database layer.

The development process

We used regular expressions in python to create a list for accession number, protein product, gene and map. We knew that some parameters had its own challenges to extract due its length, unnecessary characters around a relevant piece of data made it slightly complex to extract the exact piece of data. This included finding joins that had a format of e.g. join(516..1077,1480..1605,1977..2761 with multiple start variations. The most challenging was to extract the origin and eventually the group decided to use Bio python for this data type. The group spent lot of time setting up the database layer, especially extracting the origin and this had slightly delayed us from making a start with the business logic layer. We had a rough timeline of two weeks dedicated to each layer and having completed all the dummy API successfully, allowed us to progress onto business logic layer without fully completing the database API.

The business logic API layer was less time consuming to build but involved complex data cleaning, understanding exactly how restriction enzymes work to obtain codon coverage frequencies and percentages.

In the meantime, we also populated a basic HTML script that ensured the front end is connected and pulling all the data across successfully. In the search.cgi and listall.cgi I have added aesthetics that improved the overall appearance of the detailed page with regards to table styles, header colours and page alignments by converting basic HTML code to CSS. The style sheet biocom2.css was completed in a way to organise and improve aesthetics of the layout of the first page. The file index.py was put together by me, pointing towards the style sheet for aesthetics, getting text for the header and body and table for searching each entry.

Code testing

We tested each variable in all layers together and experienced success and failure and progressed in a way that we all understood the code and ensured all layers were connected and working. Since we made a start with the business logic layer and front end while we were still working on the database layer, it gave us an opportunity to recognise more variables that required extracting from the database layer. We continually checked each variable pulling through all the layers to front end. For front end any new change that I made in files search.cgi, listall.cgi or index.py were tested each time to ensure any new code didn't result in an internal server error.

Known issues

Extracting full origin sequence was one of the major issues we encountered, as mentioned previously we used Bio python to overcome this issues. This issue persisted further for extracting data from the GenBank file for other entries due to numerous variations in entries for different accession numbers. Also I had issues with pushing and connecting on to the correct branch of the GitHub. This was resolved by creating a personal access token that requires authentication each time I had to push files into GitHub.

What worked and what didn't - problems and solutions

There were slight issues at the beginning to create a git repository successfully as some of the team members hadn't used Git previously. But with the help of other team members and collaborators each team member managed to successfully connect to Git and access their own branches. The group slightly struggled to understand the variable coding sequence (CDS) with regards to getting a list and relating this to GenBank data page. But after research and following detailed conversations we all understood and came to a conclusion of what this means and how to display it in the front end.

Towards the end I also had an issue with Github, not accepting my files when trying to push and leading to loss of code for all the front-end work that was carried out. There was a back up from an earlier file we were working the week before, so I had to start again from there and lost some aesthetic features that I had put in originally. We had to remove my branch and add it again in order to push the changes through. Also we had spent a lot of time working on other layers and this meant I only had very limited time to work on the aesthetics of the front end.

Alternative strategies

We possibly could have started to work on multiple layers at the same time so that front end and alternative front end members would have had slightly more time to improve the quality of HTML and aesthetics. However, if we hadn't work collaboratively but instead, individually, it could have potentially taken us longer to troubleshoot and to get each layer working and connected.

Personal insights

I come from a clinical and a biological science background and been able to finally put together all my learning skills from this MSc and to use it in one place was very rewarding. I strongly believe that working collaboratively in the way we did made a huge difference in achieving the end product. The group had excellent communication skills, work ethic and acceptance of limited programming skills. I am very appreciative of how hard everyone worked, contributed and achieved this group project despite working remotely majority of the times.

Since majority of the time we all worked together, there were times we felt quite frustrated for not achieving or speeding things up but everyone managed to motivate each other and keep the group momentum going. I thoroughly enjoyed developing the front end working with HTML and CSS, I only hoped to have had more time to introduce some JavaScript to make it more complex and aesthetically appealing.

Appendix 1

Group meeting log:

- 24.03.2022 Group met to discuss the use CGI scripts and went through Practical 2 from course to get more familiar with how to write python code, extract data from a form
- 27.03.2022 – Data base: use of regex to create lists for each variable
- 31.03.2022 – Data base: use of regex to create lists for each variable
- 06.04.2022 – Front end: group worked together to get the CGI and API to work for the first time by using accession number
- 07.04.2022 – Data base: use of regex to create lists for each variable
- 12.04.2022 – Data base: use of regex to create lists for each variable
- 14.04.2022 – Business & front end: working on search cgi
- 19.04.2022 - Business & front end: working on search cgi
- 21.04.2022 - Business & front end: working on search cgi to create table to include all codons in a more presentable format
- 23.04.2022 - All layers: met at Birkbeck library to finalise the business layer and work on front end
- 02.05.2022 - All layers: met at Birkbeck library to finalise all layers and work on front end

Appendix 2

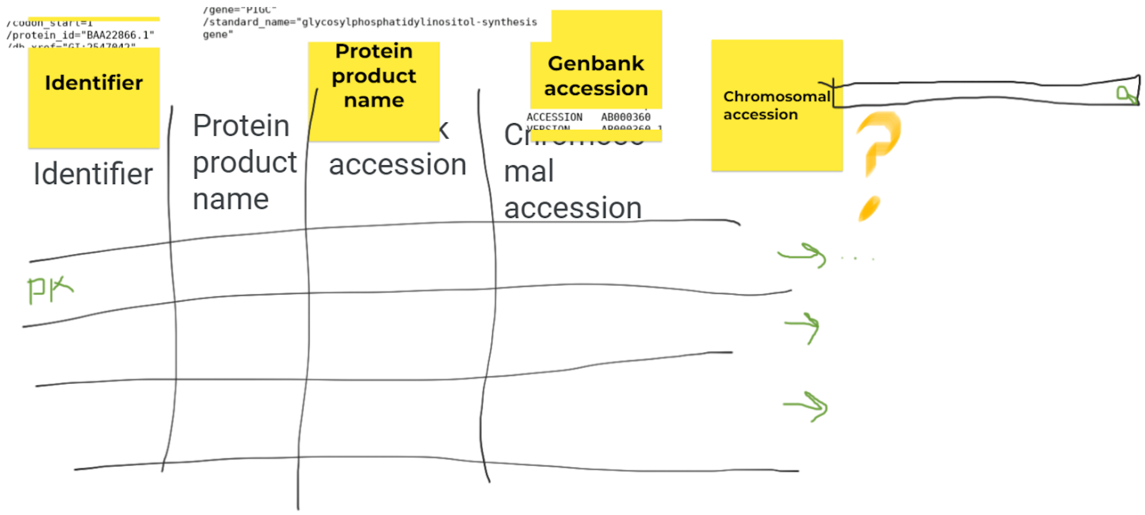

Figure 1: Front end- Listing all the variables that will appear on the list all table.

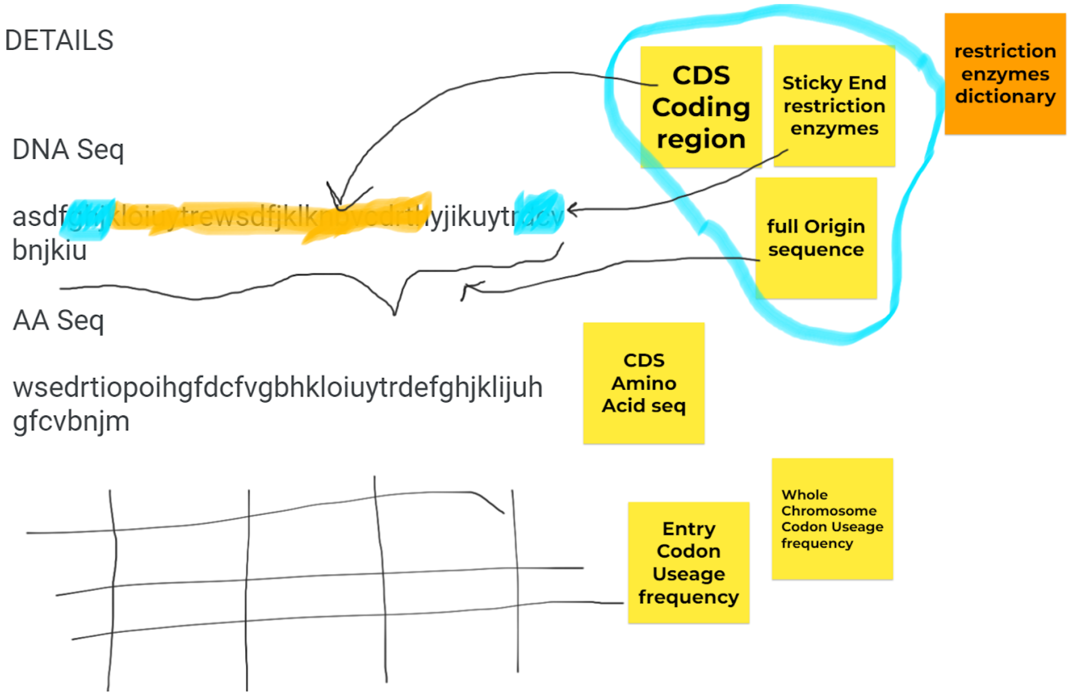
DETAILS
DNA Seq
asdfghjkloiuytrewsdfjklknjvcdrtyyjikuytrdcv bnjkiu
AA Seq
wsedrtiopoihgfdcfvgbhkloiuytrdefghjklijuh gfcvbnjm
CDS Coding region
Sticky End restriction enzymes
restriction enzymes dictionary
full Origin sequence
CDS Amino Acid seq
Entry Codon Useage frequency
Whole Chromosome Codon Useage frequency

Figure 2: Understanding the relationships between each variable.

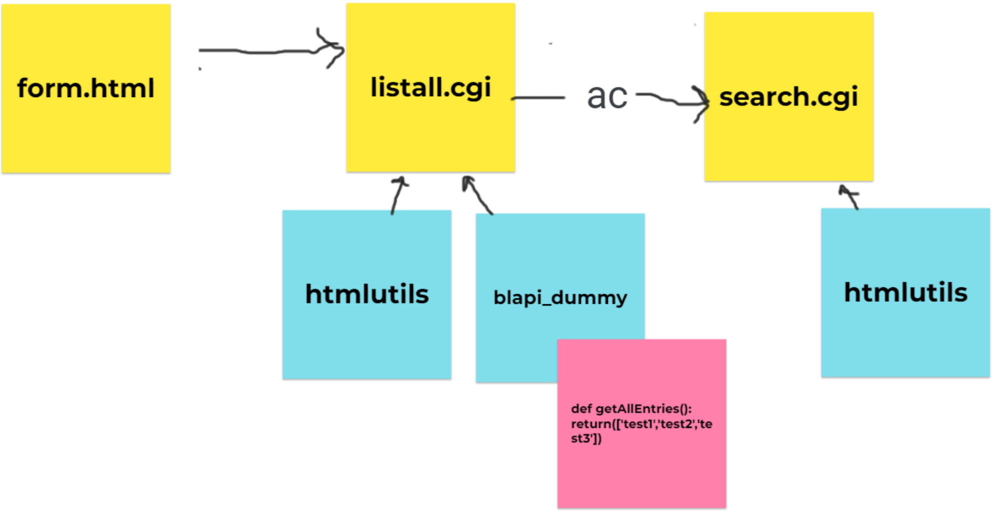

Figure 3: Front end: how each file is linked and pulling data from one another.

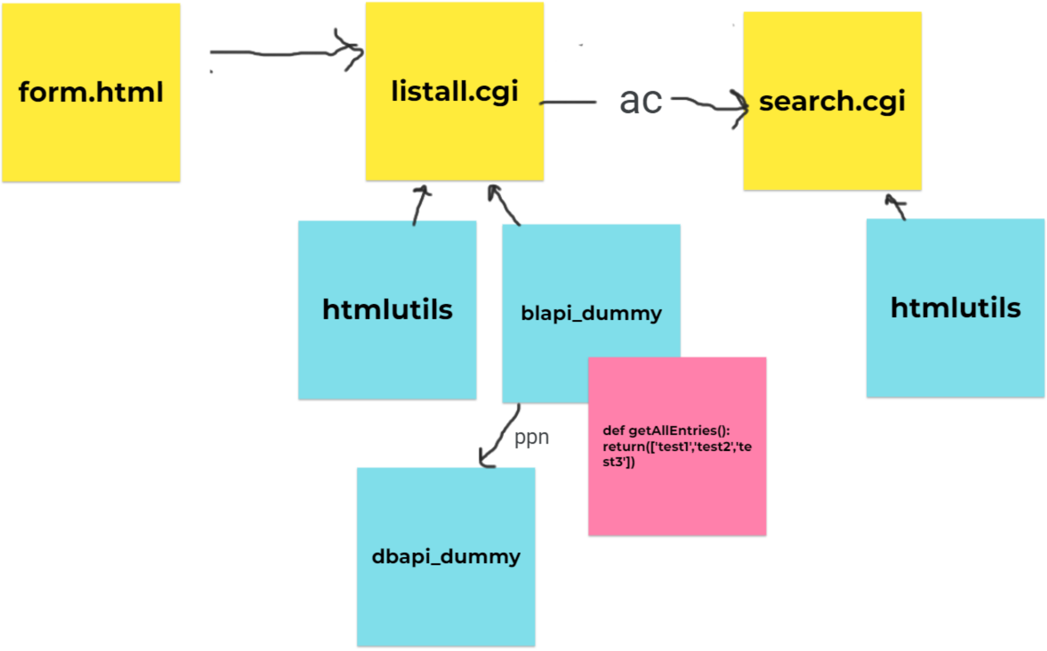

Figure 4: Front end: how each file is linked and pulling data from one another.