

Approach to the project

Interaction with the team

Our project is based on chromosome 21 and comprises of three layers: Database Layer, Logic/Business layer and the Front end. As soon as we received the instructions and our chromosome, we divided the work based on everyone's strong points and decided on a set of goals and deadlines.

Overall project requirements

Starting from the webpage requirements, we determined which data is needed to be retrieved from the database and which should be further processed by the middle layer. After some confusion regarding the APIs and the definitions, which made the progress stagnate for a short time, we all agreed on a skeleton structure of the code and the functions needed by each layer from other layers.

Requirements for my contribution

The layer that I chose to work on is the logic tier which contains the "business rules" that take requests from the interface, extract data from the data tier and perform any needed processing of the data.

The requirements I came up with include:

1. Retrieve the DNA sequence, amino acid sequence, CDS coordinates, protein product name and gene accession name from the database layer
2. Find the coding and non coding DNA sequence using the CDS coordinates
3. Use the DNA sequence to generate the codons and workout the frequency
4. Identify sticky-end restriction enzyme sites in the genomic DNA using the CDS coordinates

Performance of the development cycle

The development cycle worked well within our group. Despite the lack of physical meetings due to personal circumstances, we kept constant contact through messaging tools. This allowed us to get support from each other when any of us got stuck. Secondly, this allowed us to keep accountable and check the progress of each of us.

The development progresses

While I had personally devised a strategy for developing my layer, I discovered that the best method for me was to set an initial plan which would be changed and improved as the coding process advanced. This way, by planning and coding at the same time I was able to make sure I was going in the right direction and not continue and get stuck further along the process to eventually have to go back a few steps and re-write a big piece of code. This definitely saved time and effort.

Code testing

While the DB code was still being developed, I wrote the BL-api code using a dummy version of the DB layer, from which I called specific methods. I tested my code initially by printing the results of each function as I wrote it and checking the output. After the first round, and as the code became more complex, I used pytest, one of python's libraries, which made the testing process easier.

Known issues

One of the known issues at the moment is the function to display the table for codon frequencies in all entries in the database. Perhaps the most important issue is connecting the BL api to the DB api because of a misunderstanding in the ways we thought about organising the way the entries are retrieved from the database. The way the DB api was written would now require me to rewrite a big part of the api, which given the time constraints is not possible right now. However, the code for the BL api can be tested using the dummy.py version of the DB layer that I created.

What worked

In my opinion, having the project split into different layers with each member being allocated a different layer meant that each one of us had more or less the same amount of involvement in the project, while also working on the layer that we chose depending on our strengths and preferences.

Personal Insights

What I learned from this project experience is that communication is the most important key to developing a good, working end result. What is as important is setting an initial plan which all members of the groups should agree to and follow. On the same note, I also think all layers should be written following the same programming practice/convention (for example, to decide on a convention about how to set class names, method names and attributes). Another thing I learned is that asking for help is right and we cannot know everything. A lot of good information and help can be found online, on forums or platforms like stackoverflow.