



**I-BIX-DAT**

**Data Integration and  
Interaction Networks**

**Assignment**

**Dr Tomasz Kurowski**

**[t.j.kurowski@cranfield.ac.uk](mailto:t.j.kurowski@cranfield.ac.uk)**

**14 February 2025**

**[www.cranfield.ac.uk](http://www.cranfield.ac.uk)**



**Aim**

You will implement a REST API service for storing and sharing variant data for eukaryotic genomes.



## Objectives

You will develop:

1. A relational database (SQLite) capable of storing variant data.
2. A tool for parsing annotated VCF files to populate the database.



## Objectives

You will develop:

### 3. A REST API server deployable on Node.js and capable of serving the following requests:

- Listing all VCF samples (genomes) loaded into the database.
- Listing the number of variants (SNPs, InDels, or both) contained in each sample, grouped by chromosome.
- Listing genes impacted by high impact variants in a specified chromosomal region of a specific sample.
- Listing samples which contain variants that impact a specified gene.
- Visualizing chromosomes with high impact variants marked & labelled with the names of the genes they impact.



## Input data

You will be using VCF data with variants contained in resequenced tomato genomes.

Three files (subsets of variants from three different cultivars) which you will need to process are on Canvas. They come from the following publication:

Aflitos et al., Exploring genetic variation in the tomato (*Solanum* section *Lycopersicon*) clade by whole-genome sequencing. Plant J. 2014 Oct;80(1):136-48. doi: 10.1111/tpj.12616.

Format specification: <https://samtools.github.io/hts-specs/VCFv4.1.pdf>



## Annotation

The variants have been annotated using SnpEff. Note that these are the older “classic” SnpEff annotations, which use an “EFF” field instead of an “ANN” field. See the documentation:

<https://pcingola.github.io/SnpEff/snpSift/filter/>

Example:

```
SL2.50ch09      71507830      .      ATTTTTTTT      ATTTTTTTTT      62.5      .      INDEL;DP=30;AF1=1;CI95=1,1;DP4=0,0,14,15;MQ=60;FQ=
:Solyc09g092400.1.1|4|1),SPLICE_SITE_ACCEPTOR(HIGH|||Solyc09g092400.1||mRNA:Solyc09g092400.1.1|5|1)  GT:PL:DP:GQ      1/1:103,87,0:29:99
```



Homozygous (1/1) **HIGH** impact InDel variant affecting the Solyc09g092400.1 gene



## Database structure

You will need to design and implement a schema (i.e., write an SQL script which will create your tables) capable of storing all the data that you need in an appropriate form. An "appropriate" schema should make it easy to execute the queries you need – don't make it too complicated!

The SQL database schema creation script is part of the deliverables.

An ER diagram for the schema would be useful for visualising it in your technical document.





## Parsing inputs and populating the database

You will need to implement a data import tool (using the programming language of your choice) which will populate the database with the data from the VCF files.

The data import tool must be included among the deliverables.

The database file must also be included, and it must contain the full data from all three datasets. If you decide to leave any data out, justify it in the documentation.



## REST API server

You will implement a REST API server which can be deployed on localhost and accessed via HTTP (use curl for testing!).

The NPM package with your code must be included in the deliverables.

Do not include the `node_modules` directory in your submission – any dependencies should be listed in `package.json`.



## REST API endpoints

You are responsible for designing reasonable API endpoints for each of the possible requests, as well as for the formatting of the output.

The outputs shown in the following slides are examples. If you decide a different format is more appropriate, use it. Make sure you justify your design choices in the documentation.

Each of the API endpoints should be documented.



## Listing all the VCF datasets (genomes) loaded into the database

Example output:

```
[  
    "RF_001",  
    "RF_041",  
    "RF_090"  
]
```



## **Listing the number of variants (SNPs, InDels, or both) contained in each genome, grouped by chromosome**

Example output: number of InDels for RF\_001:

```
{  
    "SL2.50ch01": 1152,  
    "SL2.50ch02": 483,  
    "SL2.50ch03": 729,  
    "SL2.50ch04": 734,  
    "SL2.50ch05": 508,  
    "SL2.50ch06": 666,  
    "SL2.50ch07": 557,  
    "SL2.50ch08": 427,  
    "SL2.50ch09": 602,  
    "SL2.50ch10": 479,  
    "SL2.50ch11": 645,  
    "SL2.50ch12": 799  
}
```



# Listing genes impacted by moderate / high impact variants in a specified chromosomal region of a specific sample

Example output: genes affected by high impact variants within the first 20 megabases of chromosome 3 in RF\_041:

```
[
  {
    "position": 1068624,
    "gene": "Solyc03g006480.1"
  },
  {
    "position": 4910336,
    "gene": "Solyc03g033330.2"
  }
]
```

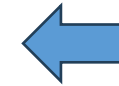
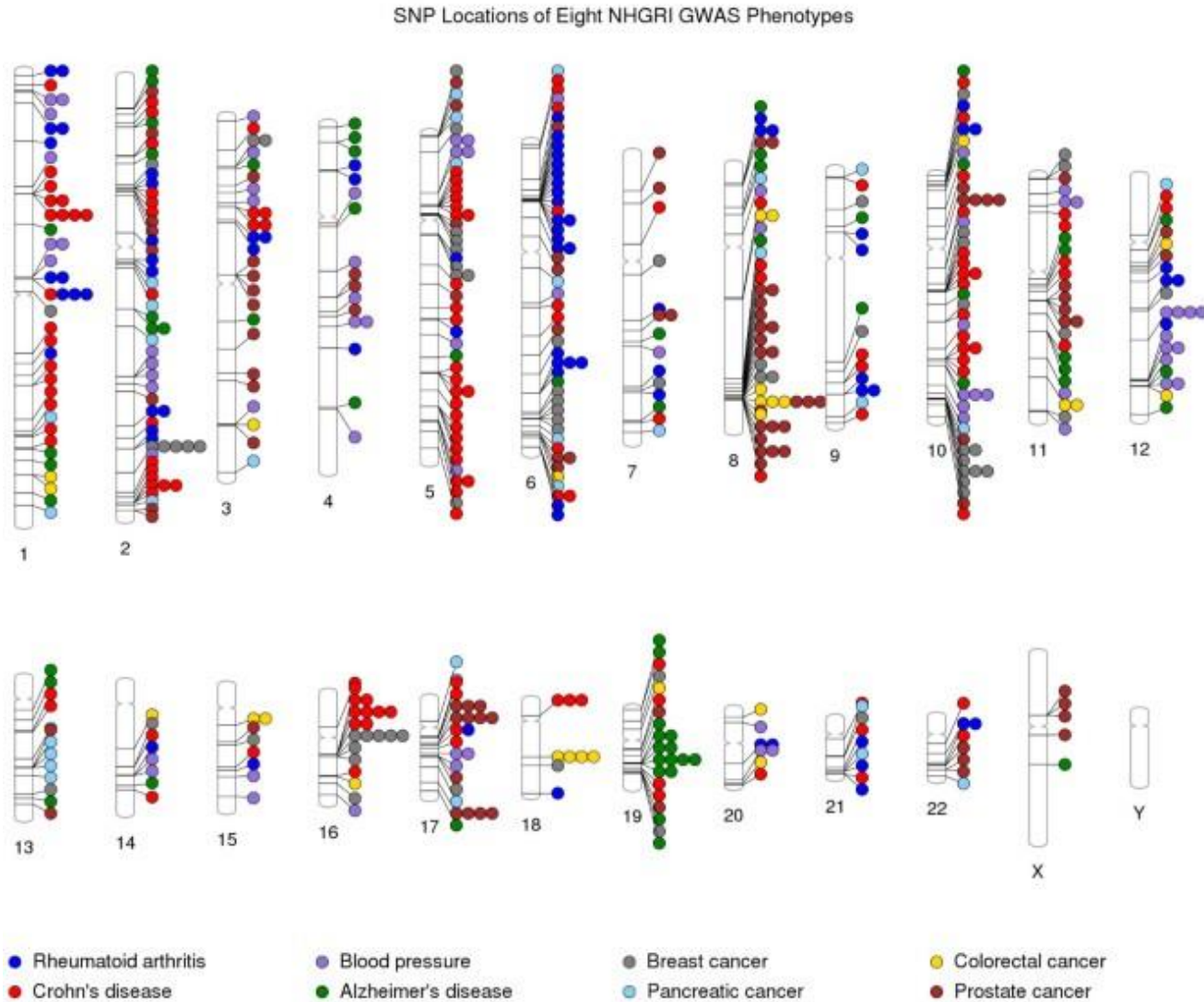


## Listing samples which contain variants that impact a specified gene

Example output: samples which contain a variant in the Solyc01g056370.2 gene:

```
[  
    "RF_041",  
    "RF_090"  
]
```

# Visualizing chromosomes with high impact variants marked & labelled with the names of the genes they impact



Example of a  
(human!)  
chromosome  
visualisation.





## Other files

You can download the full files, as well as others from the same dataset, at the Sol Genomics Network website:

- [https://solgenomics.sgn.cornell.edu/organism/Solanum\\_lycopersicum/tomato\\_150](https://solgenomics.sgn.cornell.edu/organism/Solanum_lycopersicum/tomato_150)
- [https://solgenomics.net/ftp/genomes/tomato\\_150/150\\_VCFs\\_2.50/](https://solgenomics.net/ftp/genomes/tomato_150/150_VCFs_2.50/)

They could prove useful for stress-testing your application and developing extra features.



# Marking scheme

| Specification  | Marks     |
|--|-----------|
| <b>Software (70%)</b>  |           |
| Functional SQLite database (database creation SQL script must be delivered).                                   | <b>10</b> |
| Tool for processing VCF files exists and data from the three data sets from Canvas were placed in the database | <b>10</b> |
| REST API can list datasets   | <b>5</b>  |
| REST API can report the numbers of variants  | <b>10</b> |
| REST API can list genes affected by moderate / high impact variants  | <b>10</b> |
| REST API can list samples with variants in a specific gene   | <b>5</b>  |
| REST API can visualise chromosomes with high impact variants & impacted gene labels                            | <b>5</b>  |
| Clean, well-structured and commented code  | <b>5</b>  |
| Extra functionalities  | <b>5</b>  |
| Stability  | <b>5</b>  |
| <b>Documentation (30%)</b>   |           |
| Technical documentation - relevance, conciseness, accuracy   | <b>15</b> |
| User manual  | <b>10</b> |
| Flowcharts, diagrams, ER diagram of database   | <b>5</b>  |



# Deliverables

**Deadline:** 22 February 2025 at 23:59 (Full-time)  
8 March 2025 at 23:59 (Part-time)

**Deliverables:**

- Archive containing the REST API server project.
- Archive containing the parser tool (if separate from REST API).
- SQLite database file.
- SQL script for database creation.
  
- Documentation: Technical document and user manual.