Contents

[1. Introduction 1](#_Toc30937603)

[2. Version 1](#_Toc30937604)

[3. User Manual 1](#_Toc30937605)

[4. Specifications 1](#_Toc30937606)

[4.1 Purpose 1](#_Toc30937607)

[4.2 Scope 1](#_Toc30937608)

[4.3 Software Overview 1](#_Toc30937609)

[4.4 Infrastructure Requirements 1](#_Toc30937610)

[4.5 Functional Requirements 2](#_Toc30937611)

[5. Design 2](#_Toc30937612)

[5.1 SQL Database 2](#_Toc30937613)

[5.2 Retrieve data from websites 4](#_Toc30937614)

[5.2.1 Human Kinases 4](#_Toc30937615)

[5.2.2 Phopsphosites 4](#_Toc30937616)

[5.2.3 Inhibitors 5](#_Toc30937617)

[5.3 Population of data from csv files into database 5](#_Toc30937618)

[5.4 Queries to the databases 5](#_Toc30937619)

[5.5 Development of Website 5](#_Toc30937620)

[5.6 Data analysis 5](#_Toc30937621)

[6. Systems requirements 5](#_Toc30937622)

[7. Data flow 6](#_Toc30937623)

[8. Testing 6](#_Toc30937624)

[9. Limitations and prospective work 6](#_Toc30937625)

[10. Contacts 7](#_Toc30937626)

# Introduction

Protein kinases are enzymes that modify other proteins by adding ATP to specific amino acids in a protein. The phosphorylation of these specific sites in the protein called phosphosites can modify the enzymatic activity and lead to the alteration of biological processes in the cell.

When the activity of kinases is deregulated, the modification that they produce in the phosphosites can cause diseases such as cancer or metabolic and inflammatory disorders. Several drugs have been tested and develop to inhibit the activity of these kinases involved in diseases.

The recently developed software JACKY centralises the information on human kinases and their phosphosites and inhibitors in one single tool that allows the user to find detailed information on kinases and to upload datasets to calculate the activity for research results.

# Version

This is the first version of JACKY software developed and implemented in February 2020.

# User Manual

WE CAN USE HERE A REFERENCE TO AN ADDITIONAL PDF FILE WITH SCREENSHOTS FROM OUR WEBSITE TO GUIDE THE USER ON THE USE OR THE APP (IT WOULD BE LIKE A QUICK GUIDE)

# Specifications

I NEED TO WORK ON THIS BECAUSE SPECIFICATIONS AND REQUIREMENTS MAY BE IN THE SAME SECTION. MY COLLEAGUE HAD A SECTION FOR SPECIFICATIONS STATING WHAT THE SOFTWARE NEEDED TO DO SUCH AS USING IMPORTING CSV FILES AND THE ACTION TO DO TO FULFILL THAT.

## Purpose

## Scope

## Software Overview

## Infrastructure Requirements

## Functional Requirements

Able to do wahverther, how

Consolidate information form different wesites

Create a database

Doing queries

Interaction with the user

Data analysis

# Design

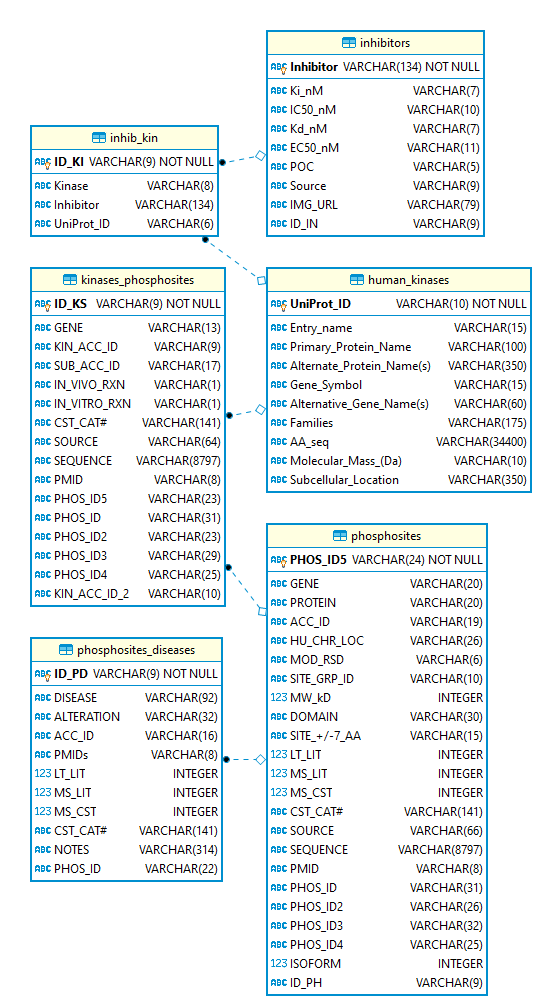
## SQL Database

Rather than using multiple csv files as a source for the application to deal with data, JACKY uses a relational database management system (RDBMS) and structure query language (SQL). The advantage of using a SQL database is the easy way of managing huge amount of data in related tables and retrieving data from them performed by using simple commands and queries.

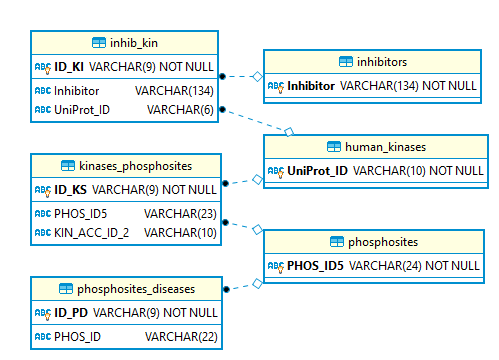
SQLite (<https://www.sqlite.org/>) is the tool that has been used to create a SQL database for different reasons. It does not require of special configuration for it use, interacts directly with files stored in the computer without the need of a server and it is free to use for any purpose.

The schema of the database to work with was created by SQLalchemy (<https://www.sqlalchemy.org/>, see file “kinases\_schema\_v3.py” for more details). This tool provided the Pythonic language to interact with SQL databases and its versatility allowed to connect the queries to the database into Flask, used for the development of the website (see section 5.5).

The design of the database consists of 9 tables, organised in object or classes in according to the structure of SQLalchemy. Three of this tables (human\_kinases, inhibitors and phosphosites, Figure 1) corresponds to the parent tables which primary keys are connected to foreigh keys on the children tables (kinases\_phosphosite, inhib\_kin and phosphosites\_diseases, Figure 2). The relationships between each parent table and its child table are of the type one-to many.



**Figure 1**. Entity Relationship (ER) Diagram displaying the model for the tables used in the database. Image created using DBeaver 6.3.2.



**Figure 2**. ER Diagram displaying the primary and foreign keys which define the relationship among tables in the database. Image created using DBeaver 6.3.2.

## Retrieve data from websites

TO WRITE:

HOW THE INFORMATION HAS BEEN COLLECTED, WITH DESCRIPTIONS OF THE SCRIPTS IF APPLICABLE AND REFERENCE THE NAME OF YOUR FILES USED TO RUN THE SEARCH. WHY THE SEARCH WAS ORGANISED IN THAT WAY?

WEBSITES WHERE THE INFO WAS RETRIEVED AND WHEN (IF THERE IS A VERSION FOR THE WEBSITES WOULD BE COOL, OTHERWISE THE DATE. THIS IS IMPORTANT TO LET THE USER KNOW HOW UPDATED THE DATABASE IS BUT ALSO TO ALLOW OTHER PEOPLE TO USE THE SCRIPTS AND REPRODUCE OUR SEARCHS. IF WEBSITES HAVE BEEN UPDATED AFTER OUR COLLECTION OF DATA THE DATA FOUND BY OTHER PEOPLE COULD BE DIFFERENT, REASON WHY WE NEED TO SPECIFY DATES/VERSION)

IF DATA WAS COMBINED FROM DIFFERENT SOURCES TO CREATE CSV FILES, PLEASE SPECIFY IT IN THE DOCUMENTATION. IF ANY PARTICULAR FILTER HAS BEEN APPLY TO GET THE FINAL INFO IN THE CSV FILES, DOCUMENT IT.

SAME CONCEPTS COULD BE APPLIED FOR DATA ANALYSIS, FLASK… WHY WE ARE USING FLASK RATHER THAN OTHER TOOLS…

### Human Kinases

### Phopsphosites

### Inhibitors

## Population of data from csv files into database

Pyhton files (it can be done with sql command lines as well but it is better and more “bioinformatics” to stick to Katie´s work, in my opinion)

## Queries to the databases

SQLalchemy queries

## Development of Website

Flask,

## Data analysis

# Systems requirements

AWS Free Tier

Place to store the database

List of all the software used

THIS TABLE IS FROM THE ENVIRONMENT THAT I CREATED TO RUN EVEYTHING. I THOUGHT IT WOULD BE JUST PYTHON, FLASK, ETC… BUT IT SEEMS IT IS EVERYTHING LISTED BELOW. CHECK YOURS BECAUSE WE SHOULD HAVE THE SAME INFO

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Version | Build | Channel |
| ca-certificates | 2019.11.27 | 0 |  |
| certifi | 2019.11.28 | py37\_0 |  |
| click | 7.0 | pypi\_0 | pypi |
| flask | 1.1.1 | pypi\_0 | pypi |
| flask-wtf | 0.14.2 | pypi\_0 | pypi |
| itsdangerous | 1.1.0 | pypi\_0 | pypi |
| jinja2 | 2.10.3 | pypi\_0 | pypi |
| markupsafe | 1.1.1 | pypi\_0 | pypi |
| numpy | 1.18.1 | pypi\_0 | pypi |
| openssl | 1.1.1d | he774522\_3 |  |
| pandas | 0.25.3 | pypi\_0 | pypi |
| pip | 19.3.1 | py37\_0 |  |
| python | 3.7.4 | h5263a28\_0 |  |
| python-dateutil | 2.8.1 | pypi\_0 | pypi |
| pytz | 2019.3 | pypi\_0 | pypi |
| setuptools | 44.0.0 | py37\_0 |  |
| six | 1.14.0 | pypi\_0 | pypi |
| sqlalchemy | 1.3.13 | pypi\_0 | pypi |
| sqlite | 3.30.1 | he774522\_0 |  |
| vc | 14.1 | h0510ff6\_4 |  |
| vs2015\_runtime | 14.16.27012 | hf0eaf9b\_1 |  |
| werkzeug | 0.16.0 | pypi\_0 | pypi |
| wheel | 0.33.6 | py37\_0 |  |
| wincertstore | 0.2 | py37\_0 |  |
| wtforms | 2.2.1 | pypi\_0 | pypi |

# Data flow

NEED TO THINK ON THE WAY TO REPRESENT THE FLOW OF THE DATA ON OUR SOFTWARE.

INPUT FROM USER > CREATIONG OF QUERIES BY SQLALCHEMY > LOCATION OF REQUESTED DATA IN SQLDATABASE > EXTRACT INFORMATION > OUTPUT.

SAME FOR UPLOADING FILE.

IT WOULD BE GRAPH WITH BOXES TO REPRESENT WHERE THE DATA GET IN AND HOW GET OUT.

# Testing

This software has been tested using Google Chrome, Firefox, Internet Explorer

I HAVE TRIED ON GOOGLE CHROME. WE SHOULD TRY IT ON FIREFOX AND INTERNET EXPLORER IF WE WANT TO CHECK EVEYTHING IS WORKING FINE FOR THOSE BROWSER, IN CASE THE USER HAS DEFAULT BROWSER ONE OF THESE.

# Limitations and prospective work

Not up-to-date database. For time limitations It has been impossible to create a system that allows the database been automatically updated with new records or information from the sources that JACKY feeds with. Prospective new versions of JACKY will improve that

Future ideas provide the user with their results available in a csv file that can be exported form the website.

# Contacts

JACKY was designed and the developed in January and February 2020 by the students of the MSc in Bioinformatics at the Queen Mary University of London:

**J**uan Ledesma Moreno ([j.ledesmamoreno@se19.qmul.ac.uk](mailto:j.ledesmamoreno@se19.qmul.ac.uk))

**A**nna Ruth Dearman ([a.dearman@se19.qmul.ac.uk](mailto:a.dearman@se19.qmul.ac.uk))

**C**elia De Los Angeles Colomina Basanta ([c.colominabasanta@se16.qmul.ac.uk](mailto:c.colominabasanta@se16.qmul.ac.uk))

**K**atie Tamara Skinner ([k.skinner@se19.qmul.ac.uk](mailto:k.skinner@se19.qmul.ac.uk))

**Y**utang Chen ([y.chen@se19.qmul.ac.uk](mailto:y.chen@se19.qmul.ac.uk))

Please feel free to contact us for support, feedback or any comment that you consider could be done to improve the software. Your opinion is important for JACKY.