**Packages Used and how they were connected:**

HTML/CSS:

**Bootstrap 4**: This is used for general look and feel of the web application. Bootstrap is widely used for styling of websites. We used it for styling our HTML elements and div(s). For example: to create a HTML page in several parts we used cards to display each part, allowing us to implement an information-related division on page.

**HTML tags:** These were used to define the components of each page. HTML was coupled to CSS languages to deliver a well presented application front with visual features for a more enjoyable experience.

**CSS** was used to generate professional search fields, and to work with the JavaScript and other interactive elements. For example: CSS grid was used to ensure separate cards on the page are in line. Other format and presentation aspect were also implemented using CSS.

**Limitations**

We could have looked into using W3-CSS is a CSS framework that also contains built in responsiveness. It is smaller and faster than CSS and with a few commands produced a clean result. Furthermore, this would eliminate the need to manually update CSS.

JavaScript:

Phosphokinase runs opensource JavaScript libraries to provide an interactive experience for the user, as well as providing another platform for more detailed information about protein kinases. JavaScript libraries that we have used are:

**jQuery**: jQuery is a widely used JavaScript library for introducing interaction to the website. JavaScript handles all the client-side dynamism in web applications. jQuery works as a library for very commonly used web interactions. Bootstrap also uses jQuery. We used it for implementing export functionality.

**html2canvas**: This is being used as a tool to convert the html page to a html canvas image. This image could be later used to create the pdf that is exported.

**Jspdf**: This is a library that we are using to create the pdf from the canvas image.

**Limitations**

?

Python:

**os**: This is the python library that we use for general os variables, like current working directory and creating file paths.

**flask**: Flask is a light-weight web application framework that helps us build dynamic web applications with least footprints. We have used several flask extensions to handle forms, errors and variables**.** Flask is based on Jinja 2 templating (Jinja2 works with Python 2.6.x, 2.7.x and >= 3.3) which shares a language similar to python, this made it easy to integrate and run code on the main application as well as on the HTML pages.

**werkzeug**: We use this library to handle file names. It was necessary to handle file names while uploading the files to web app.

**pandas:** pandas is used for reading data from csv or other data files. We have used it in Kinase Analysis and in database Access.

**numpy:** numpy is widely used for creating graph plots we have used this in Kinase Analysis as well.

**bokeh:** It is a modern library that helps create interactive presentation in modern web browsers, we used it for plotting the graphs in Kinase Analysis.

**Limitations**

**?**

**Usability of front end**

Home page

The web application has been designed to give the user a smooth experience flowing through out the website, connecting all parts without reaching any dead ends. From the home page all searches are displayed according to their relevance. The central based search is connected to our database backend. The kinase activity upload is connected to analysis python codes relevant for kinase activity analysis of .tsv files. At the top the genome browser is linked to an external website NCBI, which allows the user to search the whole human genome using a known database outside our own one. Also, tabs at top bar present contact and about information pages for the user to get in touch with the development team if necessary, for enquiries or collaboration on a more personal level. The link of the home page is implemented throughout the website, allowing the user to go back to start point at any stage through their experience.

Kinase search

Using a case insensitive search, user is able to search our database through selecting from our dropdown options (Kinase name, inhibitor name, phosphate name). This leads to a search page relevant to the closest match to user enquiry. User can select the search that they prefer and then see the page of information related to that selection. User then sees a page of information in either tabs format or single page. If in tab format, each tab displays the relevant information of aspects related to user enquiry search. The protein sequence of the kinase can be exported for future reference or external use. The inhibitors targeted by that kinase are linked to their own information pages. If the user wishes to see more information related to the inhibitor, the user can click as intended.

Kinase activity upload

The user is able to upload statistical data in form of tsv file to get an interactive output of statistical analysis consisting of quality of the image to be rendered which is then displayed in the browser. The user is then able to click the export link which results in an option to download the image as a PDF file to the user’s computer.  
If an incorrect filetype is uploaded, the software rejects the file and displays an error message. Further analysis could be considered for further development. The size of file upload is also another limitation. The library modules used for rendering the statistical results are currently only works with 32bit of python. Unfortunately, using 32bit python would cause memory errors when trying to upload large files (> 300MB) causing the program to crash. This was an issue which currently has not been resolved and is important to discuss as the creation of software is meaningless if it has not been coded for different platforms.

More information links

These links are present throughout the tabs where necessary for the user to see relevant websites such as Uniport of NCBI for more information such as 3D images.

Error page

Message uploads letting the user know that they have reached an unsatisfied search and the user is directed via a link back to the home page.

Limitations

There were several limitations in smooth usability for searching, such as user having to select a drop down in the search rather than having a combined search. Furthermore, while the user is searching a drop down of relevant accusable options do not become present to help navigate the user through their typing.

This application has been designed for website access and is not compatible for a phone app. Therefore, users do not have the easy accessibility of downloading our application onto their smart phones for quick access. They must instead go onto our website, taking more time to do their search.

Interactive 3D images of kinases are not presented; thus, the user has to click the link provided and further search an external website for that image. This would take the user attention away from our application.

These limitations slow down the user from getting to their ultimate destination on our application. Therefore, it is important to consider these as future developments. These are vital for our users to successfully reach their query and to keep our users from leaving this application.

Future developments

Our database was developed in short time frame is not comparable to that of major biochemistry/sequence websites. However, to develop our application further vital improvements must be made in the fields of several possible improvements, technical solutions / optimisation, general further development

* Interactivity
* Statistical analysis of files link to download graphs
* Ability to Login
* Updatability
* User interface visual experience
* Backend database design and data available
* User personalisation tools to save certain kind of search history or molecule of interest.

**Website architecture**

A screenshot of a cell phone

Description automatically generatedThe architecture of this web application is shown here in Figure 2. Here is showing the routes linking each part of the web application and their related links with each part separated by colour coding. This concise summary shows the journey from home page through every part of the application. Home has three top bar tabs for about, contact and genome browser. In the centre of the home page there is present one direct route to three search engines and one direct upload interactivity to a data analysis tool. Each search tool outputs a page with information allocated into different tabs. The export functionalities allow the user to save protein sequences and statistical analysis results for future reference.

***Figure 2: Phosphokinase web application map. Showing the routes linking each part of the web application and their related links with solid blue arrows. Export abilities shown in blue round pop ups. Each part of the web application separated by colour coding. Burgendy: statistical analysis, Green: Kinase search, Purple: Inhibitor search and beige-orange: Phosphate search.***

Presentation talk points of front end

1. Talk about the home page functionalities
2. Designing the home page – why tabs placed in this order
3. Contact us and about us for potential collaboration or enquiries – communicating on a more personal level with our users
4. Home page graphical/visual affects
5. Search combined for three separate parts: kinase, phosphate and inhibitor
6. Upload implemented for kinase activity analysis
7. User friendly and smooth flow of application
8. Limitations – genome browser, statistical analysis, combining further searches,
9. Link of the website- demonstration live - 1 minute for showing the website