Chen Xin Wang - K23008656 / Saruta Kittipattananon - K23047468 / Isabella Landgrebe - K23001579 / Aïda Tadlaoui - K23090130

London COVID-19 Statistics

I - Description of the GUI and functionality of all four panels:

1) Description of the GUI:

The Graphic User Interface (GUI) allows users to interact and explore the COVID-19 data for all London boroughs visually. The GUI provides the user with information into the pandemic's impact across London through a series of methods of analysis ranging from straight on numerical values displayed to visual mapping of how the pandemic has impacted the different boroughs. From the panels the user is able to observe and compare the impacts of the covid-19 pandemic. Furthermore, it highlights to the user how heavily hit each borough was, which allows the user to delve deeper into how each borough may have reacted to the pandemic differently. For example, some boroughs may have been more strict on COVID prevention earlier compared to others, hence the difference in number of new cases. From these different visual representations, users are therefore able to perceive the data in a multitude of different ways allowing a wider spectrum of possible uses for the data, as it is no longer limited to one format.

2) The Welcome panel (1/4):

The welcome panel is displayed upon the user running the application. This panel greets the user and prompts the user to select a date range to view the COVID-19 data for the selected dates. To select the dates the panel provides a date range selector one labeled "From" and the other "To", providing easy-to-use navigation, making sure users of all backgrounds are able to comprehend its use. The date range selector allows users to specify their choice of start and end dates for the data they want to view. If the selected date range is invalid (e.g., the end date is before the start date), it will notify the user to select a valid range via a red text on the panel and also a pop up alert box, making sure the user got the message that they selected an invalid date range. Thus the user will not be able to proceed. However, once the user has selected a valid date range, they can then move on to the next panel by using the forward or backward buttons, which are also clearly labeled.
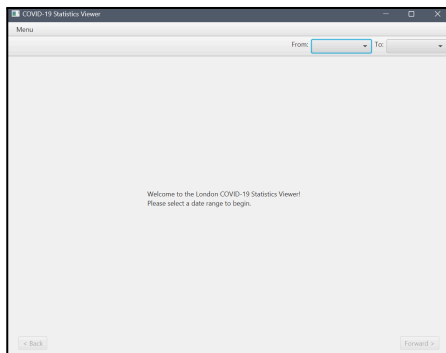

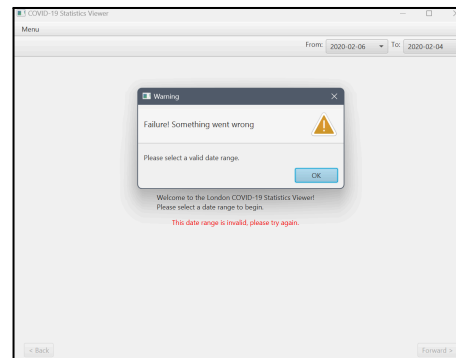*Figure 1: Initial startup page when application is open*

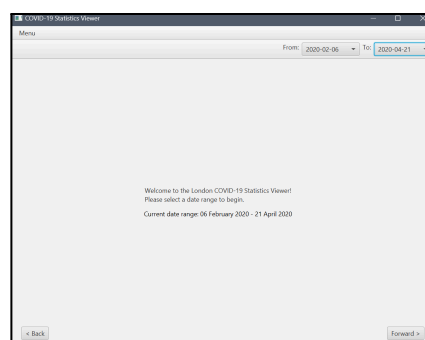*Figure 2: Page when invalid date range is inputted*

Figure 3: Page when valid date range is inputted

Chen Xin Wang - K23008656 / Saruta Kittipattananon - K23047468 / Isabella Landgrebe - K23001579 / Aïda Tadlaoui - K23090130

3) The Map panel (2/4):

Moving forward from the Welcome panel, the Map panel is then displayed to the user, revealing an interactive polygon map of the boroughs of London. The boroughs are colour-coded based on the COVID-19 death rates in each borough over that period of time, thus providing a visual representation of the impacts of the pandemic across different areas of London, giving the user a general list of which boroughs were more heavily impacted than the others. We have included a key to allow the user to understand the colour coding: for example, if the boroughs appear gray, it means that there is no known data across their selected date range for the death rates of those boroughs. Whilst the darker the color is the higher the death rate for that borough. The users are also able to select a specific borough by clicking on it to view a detailed spread of statistics related to the "new COVID-19 cases", "deaths", and "mobility changes" for their desired selected borough over the specific dates. On this pop up window of the spread of data, the user can sort the data by range of options, which is displayed via a drop-down menu. Additionally, users can continue to change the start and end dates even when on this panel. When the selected dates change the map colour and statistics also change accordingly. However, if an invalid date range is selected, a pop-up alert box will appear informing the user to reselect.
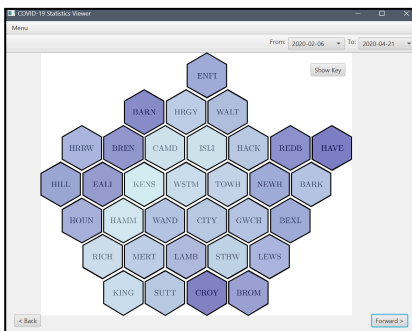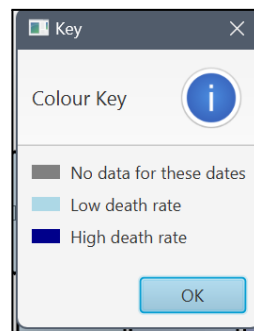

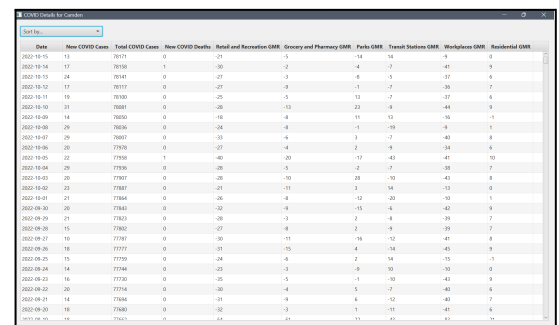*Figure 4: Map panel page*

Figure 5: Map key

Figure 6: COVID statistics for selected borough (Camden)

4) The Statistics panel (3/4):

After the Map panel lies the statistics panel, which displays various sets of COVID-19 related data, such as, "average mobility changes in different sectors", "total deaths", and "average total cases". The data is labeled big and bold, making it visually simpler for users to understand what the data on each slide represents. The user can navigate through the slides of statistics via using the buttons on both sides of the slide, allowing easy access to all the data. Like the Map panel, the statistics on the panel will similarly change depending on the date selected and users are also able to update the dates whilst on the panel. The invalid date selection consequences are also identical to those of the Map panel.
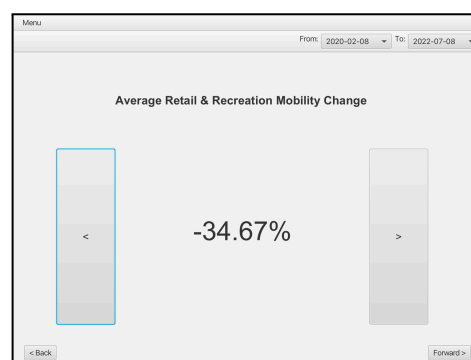

Figure 7: Statistics panel page

5) The Graph panel (4/4):

The final panel of this COVID-19 application is the Graph panel, which offers a more physical representation of the boroughs' data. It draws out graphs that display COVID-19 data trends over time, which includes "new death cases", "total deaths", "mobility trends", and "rates of case changes". The default graph displayed is the "Covid-19 Cases Over Time" for all boroughs for the selected date range. The graphs can be updated to only display data for a specific borough or view data aggregated across all the boroughs. The user can also change what data they want to view displayed on the graph, all of

which are done on a drop down menu. The graphs will also change depending on the dates selected and the user can likewise update the date whilst on this panel. In addition to this, when a specified borough is selected, the users are then given an additional function of a button labeled "Show on Map". Once this button is clicked, it opens up a web browser and takes the user to the exact location of the borough on the map, providing more detail of the location, giving users the chance to explore further. It also provides a covid prevention pop up window informing users how to avoid catching it.
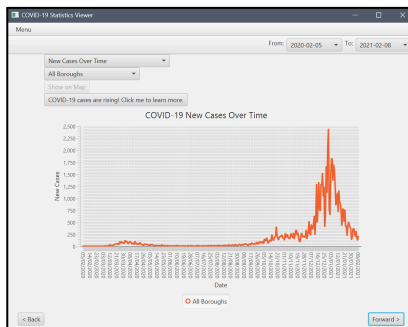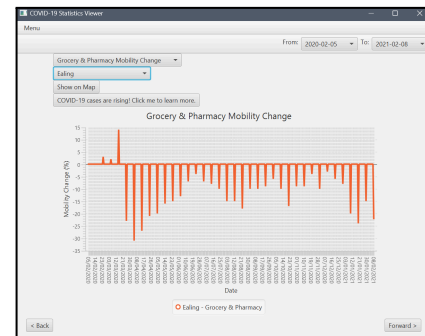

Figure 9: *Graph panel*
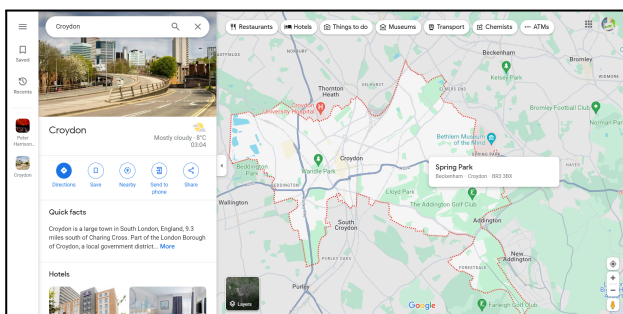

Figure 10: *Specified boroughs selected*

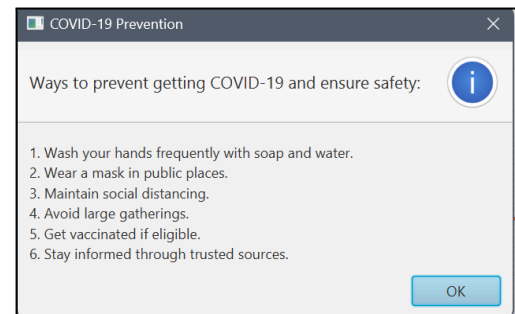
Figure 11: *"Show on map" clicked result*

Figure 12: *COVID-19 Prevention pop-up window*

II - Description of the Unit tests:

The unit test is created to test the *StatisticsPanel* class, to test whether it is correctly calculating the COVID-19 statistics based on provided data - especially including data that has empty values (to ensure that it is correctly using these values). Before each test, *@BeforeEach* is used to create an instance of the *StatisticsPanel* class so that it can be tested. There is also a mock data list of COVID-19 data that includes empty values. The test simulates the process of updating the statistics panel using this mock data list. We have also calculated the expected values by hand using the mock data list, to display the expected values. The test then checks the *StatisticsPanel* calculated values to the hand-calculated to ensure that the class is correctly calculating them. The test provides an error message to reveal if any of the statistics' calculations is incorrect. This verifies the functionality of the statistics class to improve accuracy and reliability of the application.

III - Student Contributions and Code Implementation:

- **Isabella Landgrebe :**
  - ➔ Application Window
    - ◆ *Create the application GUI*
  - ➔ Welcome Panel
    - ◆ *Create the welcome panel*
    - ◆ *Create the data selection*
    - ◆ *Create the Menu bar and Exit button*
  - ➔ Statistics Panel
    - ◆ *Ensure that the panel buttons are disabled when an invalid date is selected*
    - ◆ *Fixed the statistics panel*
  - ➔ The Map Panel
    - ◆ *Create the map panel*

- ◆ *Calculated the coordinates for the polygons*
- ◆ *Implemented the color-changing polygons depending on death rate*
- ◆ *Add the key button for the colors of the polygons*
- ➔ Unit Testing
  - ◆ *Created the unit test for the StatisticsPanel class*
- ➔ Challenge Tasks
  - ◆ *Add the ability to select a specific data to the graph*
  - ◆ *Created the Total Deaths Over Time, Mobility Change and Rate of Case Change graphs*
  - ◆ *Ensure that the date, borough and graph can be changed and the graph is correctly updated*
  - ◆ *Ensure that only a certain number of data points are shown (even for a large range of data)*
- ➔ Report
  - ◆ *Description of the GUI and panels*
  - ◆ *Commented on the CovidStatsApp, StatisticsPanelTest, MapPanel, BoroughMapper classes*

- **Saruta Kittipattananon :**
  - ➔ Welcome Panel
    - ◆ *Create the data selection*
    - ◆ *Create the buttons to the side of the screen*
    - ◆ *Implemented the buttons so that they are disabled if an invalid date is selected*
  - ➔ The Map Panel
    - ◆ *Polygons can be selected and show the data available for each table. The table can be sorted by each of the data*
  - ➔ Statistics Panel
    - ◆ *Fixed the statistics panel*
  - ➔ Challenge Tasks
    - ◆ *Fixed the layout of the graph to fix the bugs*
    - ◆ *Ensure that the date, borough and graph can be changed and the graph is correctly updated*
  - ➔ Report
    - ◆ *Description of the GUI and panels*
    - ◆ *Commented on the CovidStatsApp, MapPanel, BoroughMapper classes*

- **Aïda Tadlaoui :**
  - ➔ Welcome Panel
    - ◆ *Show the current date range selected*
    - ◆ *Create the alert message that is displayed when the date range is invalid (works on all panels)*
  - ➔ Statistics Panel
    - ◆ *Create the statistics panel*
    - ◆ *Create the buttons*
    - ◆ *Implement four statistics for the selected period*
    - ◆ *Ensure statistics are updated when the date range changes*
  - ➔ Challenge Tasks
    - ◆ *Add the ability to select a specific borough to the graph*
    - ◆ *Fix the graph window*
    - ◆ *Add to the graph panel the "Show on map" button*
  - ➔ Report
    - ◆ *Layout*
    - ◆ *Commented on the StatisticsPanel class*

- **Chen Wang :**
  - ➔ Welcome Panel
    - ◆ *Edited update date range function*
  - ➔ Challenge Tasks
    - ◆ *Idea of panel*
    - ◆ *Create the graph panel*
    - ◆ *Created graph New Covid Cases Over time*
    - ◆ *Created a more info button*
  - ➔ Report
    - ◆ *Report layout*
    - ◆ *Description of the GUI and panels*
    - ◆ *Commented on the GraphPanel classes*