London COVID-19 Statistics

Coursework 4

Jiangjing, Xu
21134158
Jiangjing.Xu@kcl.ac.uk
Hongyuan, Zhao
22022013
Hongyuan.Zhao@kcl.ac.uk
Lei, Ding
21029011
Lei.Ding@kcl.ac.uk

Base Tasks

Our JavaFX application, centered on presenting COVID-19 statistics, was crafted through a collaborative process, with team members initially designing individual GUI panels. In the project's final stages, we opted for a streamlined approach by integrating all functionalities into a singular main GUI, facilitated by the `AppController`. This strategic shift was aimed at simplifying user interactions and bolstering the application's unified feel.

The `AppController`, embodying the MVC pattern, manages the application's initiation, including data loading and user interface setup, via its `initialize()` and `loadCovidData()` methods. This setup ensures the accurate depiction of COVID-19 statistics aligned with user-selected preferences, thereby enhancing the app's usability and reliability.

A noteworthy feature is the application's menu bar, which is deliberately simplified to include essential functionalities like a "Close" option for exiting the application and an "About" option for panel information. This simplification is achieved by leveraging a single `AppGUI` FXML file that incorporates various panels like `statisticsPanel`, `welcomePanel`, `mapPanel`, and `LineChartPanel` within a `StackPane`. This method, using `<fx:include>` tags for modular inclusion of panel FXMLs, significantly reduces the complexity associated with managing disparate GUI components.

This consolidation into one comprehensive FXML file signifies our dedication to providing a streamlined, user-friendly interface. It reflects our commitment to enhancing the overall user experience by presenting a cohesive platform for easy access to critical COVID-19 statistics, emphasizing our application's utility as a coherent and accessible informational resource.

Panel 1: Welcome

The Welcome Panel, controlled by the WelcomePanelController, is the user's first interaction with our JavaFX application, offering a quick introduction and tips for using the application. It utilizes FXML for layout and CSS for styling, presenting information in a clear and user-friendly manner. This panel not only welcomes users but also guides them on how to navigate the app's features, emphasizing ease of use from the outset.

Panel 2: The Map

Inthe Map Panel within our JavaFX application, the MapPanelController employs a series of methodologically crafted functionalities that underscore its efficiency and user-centric design.

The panel's interactive visualization is powered by specific methods such as updateMapVisualization(LocalDate startDate, LocalDate endDate) and updateBoroughButtonStyles(Map<String, Integer> deathCountsByBorough). These methods ensure that the map's visual representation is both dynamic and data-driven, allowing the colors of the borough buttons to change according to the selected date range and the severity of COVID-19's impact.

User interactions are handled through the handleBoroughClick(ActionEvent event) method, which captures clicks on borough buttons to display detailed COVID statistics. This method collaborates closely with showBoroughDataWindow(String boroughName, List<CovidData> boroughData), showcasing our commitment to providing detailed and accessible information. These interactions are seamlessly integrated, thanks to the preparatory work of findButtonForBorough(String boroughName), which associates each borough button with its corresponding data.

Moreover, the setMapPanelAnalyzer(MapPanelAnalyzer mapPanelAnalyzer) method establishes the foundation for data analysis and visualization by linking the controller with the MapPanelAnalyzer, ensuring that the displayed information is accurate and up-to-date.

Panel 3: Statistics

The Statistics Panel in our JavaFX application showcases the integration of StatisticsController and StatisticsAnalyzer for dynamic COVID-19 data analysis. The StatisticsAnalyzer applies methods like calculateAverageMobility and calculateTotalDeaths to analyze mobility trends and mortality impact, offering users detailed insights into pandemic patterns. The StatisticsController enhances user interaction by enabling data filtering through updateStatistics, which adjusts to user-selected date ranges for a tailored analysis experience. Navigation methods btnPrevious and btnNext allow users to explore statistical insights one at a time, ensuring clarity and focus. This design ensures that the panel remains user-centric, providing up-to-date and comprehensible statistics, reflecting our dedication to delivering an accessible and informative statistical overview of COVID-19 impacts.

Unit Testing

The StatisticsAnalyzerTest suite, utilizing JUnit, rigorously validates the accuracy of our application's StatisticsAnalyzer. It ensures precise calculations of key COVID-19 statistics, including average mobility, total deaths, and case averages, through targeted tests. This validation process affirms the application's reliability in delivering accurate COVID-19 insights, highlighting our commitment to data integrity and user trust.

Challenge Tasks

Pane 4: Line Chart

The LineChartController in our JavaFX application adeptly visualizes COVID-19 case and death trends, showcasing the application's capacity for meaningful data representation. This controller dynamically updates the line chart based on user-selected date ranges, filtering data with Java streams for precision and utilizing TreeMap to maintain chronological order of data points.

Key functionalities include the updateLineChart(LocalDate startDate, LocalDate endDate) method, which filters COVID-19 statistics for the specified period and aggregates this data into chronological series for cases and deaths. These series are then formatted and displayed on the chart, providing an intuitive and comparative analysis of the pandemic's impact over time.

Contribution

Pane 1(Welcome): Hongyuan, Zhao

Pane 2(Map): Lei, Ding

Pane 3(Statistics): Jiangjing, Xu & Hongyuan, Zhao

Unit Test: Jiangjing, Xu & Hongyuan, Zhao & Lei, Ding

Pane 4(Line Chart): Jiangjing, Xu

Report: Hongyuan, Zhao & Jiangjing, Xu & Lei, Ding