

COVID-19 Tracker

N Malarvizhi¹, Arun Kumar Dash², J Aswini³ and V Manikanta⁴

¹Professor, Dept of CSE, Vel Tech Rangarajan Dr Sagunthala R&D Institute of Science and Technology, Chennai

^{2,4}UG Student, Dept of CSE, Vel Tech Rangarajan Dr Sagunthala R&D Institute of Science and Technology, Chennai

³Professor, Dept of CSE, Saveetha Engineering College, Chennai

Email: drnmalarvizhi@gmail.com, vtu11182@veltech.edu.in, aswini.jayaraman@gmail.com, vtu15955@veltech.edu.in

Abstract— Covid-19, which is also called as “The outbreak of Corona virus” as it is commonly known as in the process of getting terminated completely, but it is never known about the future of any virus or bacteria. The prevalence of Corona Virus in our society can be successfully monitored only through proper tracking and the subsequent analysis with those tracking. The process of analyzing the tasks is completely dependent on the society experts, but they can analyze only with the tracking provided to them. The application that is being built now eases the work of the analysis experts. It will keep a track of the day-to-day cases of corona, its surges and downfall. It will show the daily changes in the form of graphs and charts which, with a better User Interface that can help not only the experts, but also general people, visualize the readings clearly by minimizing the visible details and making it more abstract suitable for smaller screen sized applications. People can check the ups and downs of the covid cases in a pictorial format through bar charts, pie charts, line graphs and other graphical diagrams. The data for all the countries will be fetched via an API call which will be updated by the API service provider itself. The data fetched will then be processed and visualized in the form of bars and charts. Apart from the daily cases, users can also view the daily deaths and recoveries. They can additionally check the total number of Covid cases till date. The total number of Deaths and recoveries can also be viewed by the user with ease. Users will be given an option to sort the type of data they want to view i.e., death, recoveries, new cases and total cases. Based on their input, the pictorial representation will be shown to them. Mobile phone users always look for an application for their comfort, thus an application which visualizes less detailing and more abstract way targeting small screen users.

Index Terms— User Interface, Covid-19, Visualization, Tracking.

I. INTRODUCTION

The title of this project "Covid-19 Tracker" in simple terms means a record of daily covid cases prevailing in the country. These records in turn helps the analyst and experts to properly understand the Covid scenario and make correct decisions for future. Furthermore, the tracking helps every individual using the app to remain up to date and aware of the future consequences. In the past, the cases turned less predictable when there was a second wave in the curve plotting the number of active cases and new cases. Though the curve of death cases did not find a clean wave second time in the graph. These incidents made the cases go unpredictable causing not only the analysts, but also the general people look into graphs everyday with less understanding of the high level graphs in mobile devices and great numbers which doesn't explain much.

Though there were many types of visualizations, more detailing and lower level graphical representations lost

views from mobile users. Mobile phone users always look for an application for their comfort, thus an application which visualizes less detailing and more abstract way targeting small screen users. The Data for all the countries will be fetched via an API call which will be updated by the API service provider itself. The data fetched from the API will then be processed and filtered. The filtered data will be used to visualize the covid cases and displayed to the users through pictorial representation in the form of bars and charts. Mobile phone users always look for an application for their comfort, thus an application which visualizes less detailing and more abstract way targeting small screen users. The main purpose of the project is to provide a better user interface to the users to view the covid changes happening daily and work accordingly. Both the common people and the covid analysts can make use of the app and keep them updated and aware.

II. LITERATURE SURVEY

In [1], the author analysed how much time people spend over small displays compared to large displays over time. Small displays are considered mobile phones, the data taken from websites which encountered API calls from devices with smaller resolutions and screen size, data visualised over various time periods. The study proved that people with smaller devices are more likely to check data related to a specific trend or fashion but often give up due to complex data over a smaller screen.

Various visualization techniques have been shown in [2] which can be applied online and offline by using static data and how it helps various professions such as policy makers, scientists, healthcare providers and the general public understand the aspects of the pandemic. These techniques came to limelight due to the pandemic and most people learnt more new terms about visualization during the pandemic. In [3], the authors use the retrieved data for Susceptible Exposed Infectious Recovered (SEIR) predictive modelling. Sentimental Analysis is done with news data by segregating into negative and positive sentiments, to understand the influence of the news on people's behaviour both politically and economically.

Audio analysis methods have been discussed by the authors in [4] using audio data sets and symptoms of people to detect the presence of COVID-19 virus from sampling of data and applying SVM. The data was crowd sourced using their application. This audio data helped people identify symptoms as during the first phase of pandemic throat infection was a major symptom for the virus. In [5], the authors discussed about the Community-level surveillance with an unsupervised symptom-based online search model is being used here. In our analysis, we compare Correlation and regression where the relationship of search frequency time series and confirmed cases or deaths can uncover symptoms or behaviours related to COVID-19.

The authors in [6] discussed about the analyses of mobility data collected by GPS and Wi-Fi by analysing the data of devices connected through public Wi-Fis through tourism boards and tracing its mobility during the pandemic. The data helped in tracking people who were possibly in contact with a vector using the device footprint logged in public Wi-Fis which were connected same time when the disease carrier was using the Wi-Fi to easily sort out the contact tracing. This made possible to track down few cases and avoid spreading.

In [7], the authors suggested various phases of COVID-19 spread control by using automated ways with the help of IOT and Internet to conduct various process like screening, testing, contact tracing, prediction, sanitation with the help of robotics and IOT instead of manual ways to increase accuracy and safety. This further helped safety of people by using robots instead of front line workers using safety guards and masks which doesn't guarantee safety in all circumstances.

III. PROPOSED SYSTEM

The architecture of the project is displayed in Figure 1. The system consists of front end and back end. The front end of this system consists of the User Interface which interacts with the user to get inputs by handling events from touch and keyboard. The user interface consists of views and view controllers which are arranged in particular layouts and order to show one over the other with specific overlays. The user interface has to filled with data, which is obtained from back end through the help of data managers which manage the data from network and database. The user interface also handles various screen sizes and resolution thereby computing the size and pixels of every frame for the specific device. Other than colour and layout, front end needs data to be filled out which is dynamic and cannot be hard coded. The data is handled by back end. The back end consists of a data manager which handles the data to be fetched from various APIs and decode the data to a specific format which the code can understand. It converts to classes and types defined earlier to the user interface and passes the data in those types to the front end. The front end utilizes this data and updates the User Interface. The data obtained from APIs after decoding is also stored in databases for further access as network calls take a lot of

time to get the response. This data is passed further to database and to the UI and received from network to the database and so on. Thus, back end manages the data and sends it to the front end through a single layer to interact to avoid multiple point of contacts between front and back end.

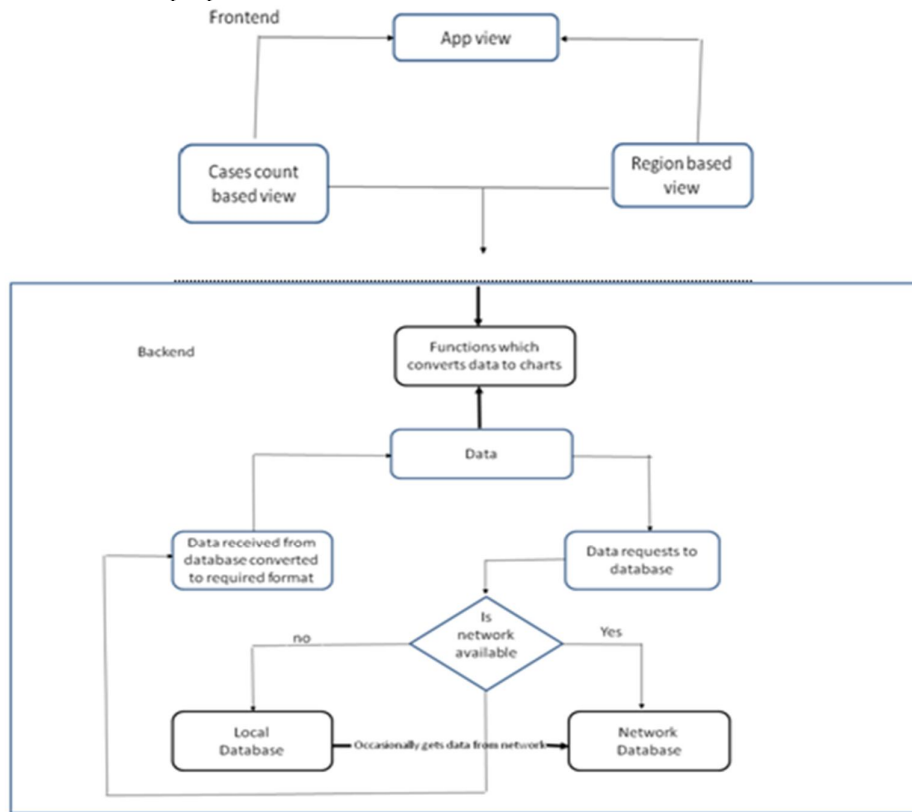


Figure 1. Architecture diagram

There are three modules in this application namely, , Domain module, Data module and Presentation module. The presentation module handles the input data received from user and converts it to function calls and passes it to the domain module. Also, this module handles all the views and visualizations presented to the user such as buttons, colours, layouts, frames and images. The domain module collects the function calls from the presentation module and passes it to the data module. The data module collects the data requests from the domain module and passes responses to the domain module the data fetched either from database or through some network calls. The final data is passed to the presentation module and displayed to the user..

A. Domain Module

The domain module consists of all types responsible for the application's back-end. It consists of the entities not limiting to the types present in the database model. These entities are used throughout the application in place of the set of data which is structured. Domain also has the implementations of the use cases which are required for the functionality of the application. It uses the help of Data module to fetch data for the specific use case and may or may not convert it to specified entities and pass back to UI. The UI gets help of these use cases to obtain the data without directly interacting with the database or API.

B. Data Module

The data module consists of all network calls such as API requests and responses, socket management and database management. The data is obtained from the network and entered to the database tables. These tables receive data from sockets also. The formatting of these data happens in this module. The formatted data is passed to the Domain module as a response.

C. Presentation Module

The presentation module performs all input and output operations with the user. This module has all the views which displays buttons, text, images, etc. These views are flooded with data obtained from Domain module. There can be inner views which also can be used with the help of the Domain module for obtaining the data. The data used here should be easy to understand as it interacts directly with the user.

D. Input Design

The input is collected through user interface items such as Buttons, selectors, touch events through the mobile's screen. The inputs are converted to use cases in the presentation module in the app and sent to domain module to collect data as a response to the input events. The inputs are logged using the default logger present in the development environment and are collected frequently while the application is used. These events are collected just in case the application crashes, the recent events are sent to the developers in the crash report to analyse the cause of the crash.

E. Output Design

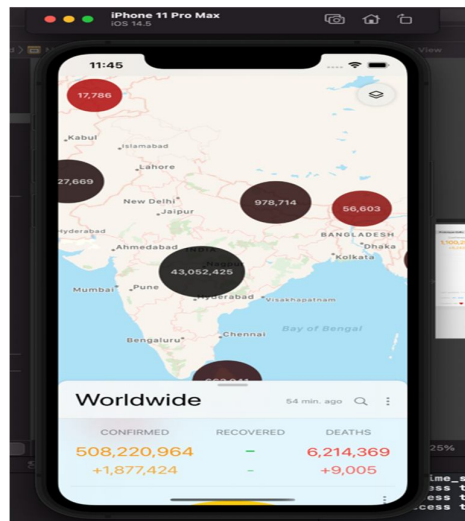


Figure 1. Screenshot of output

The output is displayed as an iOS application in iPhones. The output includes the graphical representations of the covid-19 data in bar graphs, pie-charts, half pie-charts. The data is also represented in maps showing the counts inside circular buttons, which on click displays the country's case details in a pull-over view, which can be dragged across the screen to view the details. Further, there are buttons to switch the type of data in the map visualization. A sample screenshot of output is displayed in Figure 2.

IV. RESULTS AND DISCUSSIONS

The proposed system has many UI visualizations which are simpler and easy to understand for the general audience unlike the existing system which provided with complex data which was helpful only to experts and analysts. This further dominates the need of the application by displaying live data over the application by fetching from live crowd sourced data, whereas in web we need to refresh to look over the new data. Figure 3 and 4 depicts the global and daily Covid cases respectively. Figure 5 shows the advantage of performance of Swift programming language which is used for iOS development better than react native, a leading web development language. Swift leads in performance for displaying images, graphs, maps, etc. comparatively with react native. Although, maps consume more memory comparatively with react native, iPhones are capable of handling high memory load. Swift is designed in such a way to overcome the defects of other development languages by using a LLVM(Low Level Virtual Machine) compiler, which uses a concept of Automatic Reference Counting, which saves huge memory spaces in long running applications comparatively with languages like Java and JavaScript which use garbage collector, which is comparatively slower.

The proposed system has less detailed data in graphs, charts allowing users to easily understand and get convinced over the covid cases by looking at these visualizations over the mobile phone. This system has performance advantages as it processes less data which further reduces the time and space complexity over the data which is complex in existing systems. This efficiency produced in the existing system makes this application more eligible for needs over the existing system.



Figure 2.Global Covid Cases

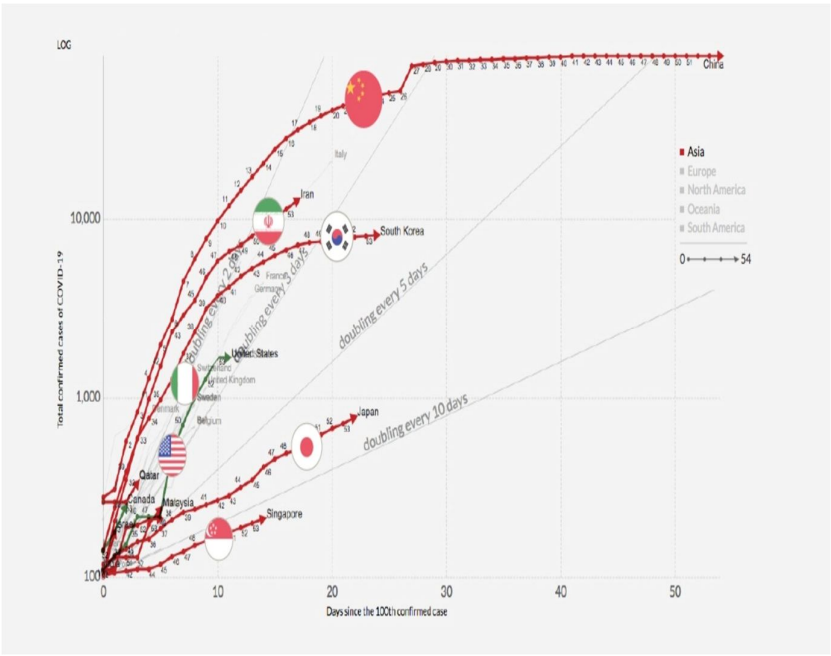


Figure 3.Daily Covid Cases

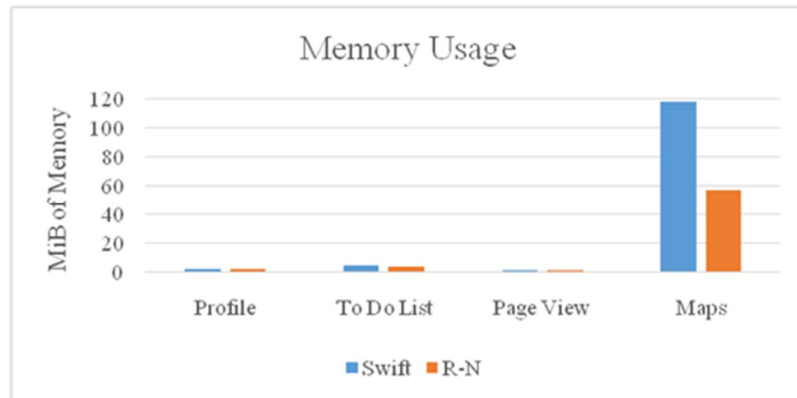


Figure 5. Memory usage comparison between react native and iOS

V. CONCLUSION AND FUTURE ENHANCEMENTS

The proposed work is mainly be used to track the Covid-19 surges. Finding, testing, isolating and treating a Covid patient is an ongoing process that is undoubtedly helping a lot of people, but, it is also vital to trace and track the day to day Covid-19 cases to keep a record and analysis from the records. The analysis done will in turn help the experts and other individuals to create awareness and increase the mental satisfaction among people. This model can be adopted for further visualization concepts such as stocks, HR management, analytics, business management, project management and health services. All industries which use graphs and other visualizations to display in mobile applications can build iOS application using simple visualization and enhance the application to be comfortable for the user which looking at the data.

REFERENCES

- [1] Brehmer, M., Lee, B., Isenberg, P., & Choe, E. K. (2018). Visualizing ranges over time on mobile phones: A task-based crowdsourced evaluation. *IEEE transactions on visualization and computer graphics*, 25(1), 619–629.J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [2] Comba, J. L. (2020). Data visualization for the understanding of covid-19. *Computing in Science & Engineering*, 22(6), 81–86.
- [3] Hamzah, F. B., Lau, C., Nazri, H., Ligot, D. V., Lee, G., Tan, C. L., Shaib, M., Zaidon, U. H. B., Abdullah, A. B., Chung, M. H. et al. (2020). Coronatracker: Worldwide covid-19 outbreak data analysis and prediction. *Bull World Health Organ*, 1(32), 1–32.
- [4] Han, J., Brown, C., Chauhan, J., Grammenos, A., Hasthanasombat, A., Spathis, D., Xia, T., Cicuta, P., & Mascolo, C. (2021). Exploring automatic covid-19 diagnosis via voice and symptoms from crowdsourced data. *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8328–8332.
- [5] Lamos, V., Majumder, M. S., Yom-Tov, E., Edelstein, M., Moura, S., Hamada, Y., Rangaka, M. X., McKendry, R. A., & Cox, I. J. (2021). Tracking covid-19 using online search. *NPJ digital medicine*, 4(1), 1–11.
- [6] Ribeiro, M., Nisi, V., Prandi, C., & Nunes, N. (2020). A data visualization interactive exploration of human mobility data during the covid-19 outbreak: A case study. *2020 IEEE Symposium on Computers and Communications (ISCC)*, 1–6.
- [7] Singh, P. K., Nandi, S., Ghafoor, K. Z., Ghosh, U., & Rawat, D. B. (2020). Preventing covid-19 spread using information and communication technology. *IEEE Consumer Electronics Magazine*, 10(4), 18–27.