IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

REMOTE ACCESS TOOL

[1]Sanskruti Punyarthi, [2]Mikil Lalwani, [3]Nilay Pophalkar, [4]Shree Samal, [5]Dr. Manoj Sabnis

[1]Student, [2]Student , [3]Student, [4]Student, [3]Professor

[1]Information Technology,

[1]Vivekanand Education Society's Institute of Technology, Mumbai, India

Abstract: In the presented work, a remote access tool has been developed utilizing the Flutter framework, with the primary objective of facilitating remote monitoring of a host device from a separate remote device. Initially conceptualized for parental control purposes, the application provides a secure and user-friendly solution for parents to actively supervise the digital activities of their child's mobile device. The tool is designed to offer real-time activity monitoring, content filtering, spam SMS detection and detailed reporting, empowering parents to enhance their child's safety and well-being in the digital age. Its capabilities extend to promoting responsible device usage and cultivating a secure digital environment for children. This abstract concisely outlines the core functionality and objectives of the remote access tool, emphasizing its potential utility in parental monitoring scenarios, while acknowledging its adaptability for broader use cases beyond the parent-child scope.

Index Terms **- remote access tool, parent - child monitoring, sms spam detection, parental control system, smartphones**

I. INTRODUCTION

In an increasingly digital world, the need for tools that empower responsible and safe device usage has never been more pronounced. With over 7 billion smartphones worldwide, representing a staggering 86% of the global population, the ubiquity of these devices underscores the urgency of addressing digital safety and responsibility [1][2]. Particularly concerning is the surge in teenage internet usage, with more than 96% of adolescents accessing the online world on a daily [2]. Our project introduces a versatile and user-friendly remote access tool developed using the Flutter framework.

This tool serves as a powerful solution for monitoring the activities of a host device from a remote location. Originally conceived with a focus on parental control, it provides parents with a means to closely oversee their child's mobile device usage, ensuring their safety and promoting responsible digital behavior. The impact of this remote monitoring tool on people's lives is substantial and far-reaching. While recognizing the transformative potential of this remote monitoring tool, we also acknowledge the importance of addressing concerns surrounding privacy, security, and ethical usage. It addresses several critical aspects of modern digital living and has the potential to positively influence various stakeholders, including parents, children.

II. AIM AND OBJECTIVES

A. Impact on Parental Monitoring

The objective is to evaluate the practical impact of the tool on parental monitoring. This will assess its effectiveness in helping parents monitor and manage their child's mobile device usage, aiming to understand any positive changes in child's behavior and parent-child relationships that result from its use. This will also help to protect the child from the increasing rate of digital crimes involving smartphones such as scams, frauds, phishing, cyber-bullying and many others.

B. Elderly People

With the sudden rise of technology in the past few decades, some people have not been able to keep up with this trend. In particular, elderly people are the ones who have been on the losing end. As smartphones have become a necessity in contemporary society, the older generation has no choice but to adapt. So the children or the caretakers people who are responsible for the welfare of these people need to keep an eye so as to prevent any mishap involving smartphones. Cyber-criminals find it easy to target such people. So this solution will help in prevention of such unfortunate events.

C. Use Cases Beyond Parental Control

While the tool is initially designed for parent-child dynamics, the objective is to explore potential use cases and scenarios where the tool's functionality could extend beyond this primary scope. To analyze its adaptability and versatility for broader applications, such as for people with mental disorders.

III. LITERATURE SURVEY

Researchers have aimed to provide a comprehensive understanding of what Remote access tools are and how they work. These studies delve into the technical details, explaining the mechanisms through which these tools allow remote control of computers [3]. These systems proposed how a computer can be controlled remotely using a smartphone. Nowadays, mobiles have a plethora of sensors which allow for a more in depth study of human behavior[4][5]. With mobiles becoming a necessity in today's society, classroom's across the world are taking efforts to reduce the dependence of students on such devices by banning them[6]. Initial studies highlight the concerns parents have regarding their children's online activities. With the increasing accessibility of digital devices, parents express the need for tools that allow them to monitor and control what their children are exposed to on the internet [7]. The key findings from studies derived through analyses reveal significant privacy concerns and regulatory non-compliance [8]. Different software applications have been developed for the aid of parents in making sure that they have control over their child to some extent while ensuring the privacy and security of their child[9][10].

IV. METHODOLOGY

A. Proposed System

The proposed system operates through two distinct applications: a host app installed on the target device and a remote monitoring app on a separate remote device. The host app, leveraging the capabilities of Flutter and various associated packages, serves as the tool for exposing critical data such as SMS, call logs,contacts, and real-time location information. Through a robust WebSocket connection and HTTP requests, the remote monitoring app gains access to this data, facilitating real-time observation and tracking of the host device's activities. Combined with the SMS spam detection model, the system will inform the parents via notification if such a spam SMS is received. This system provides a secure and efficient means for users, typically parents, to remotely monitor and oversee the digital and physical interactions of the target device. It brings together the power of Flutter's extensibility and real-time communication protocols, enhancing the user's ability to maintain a watch over the target device, thereby ensuring digital safety and responsible device usage.

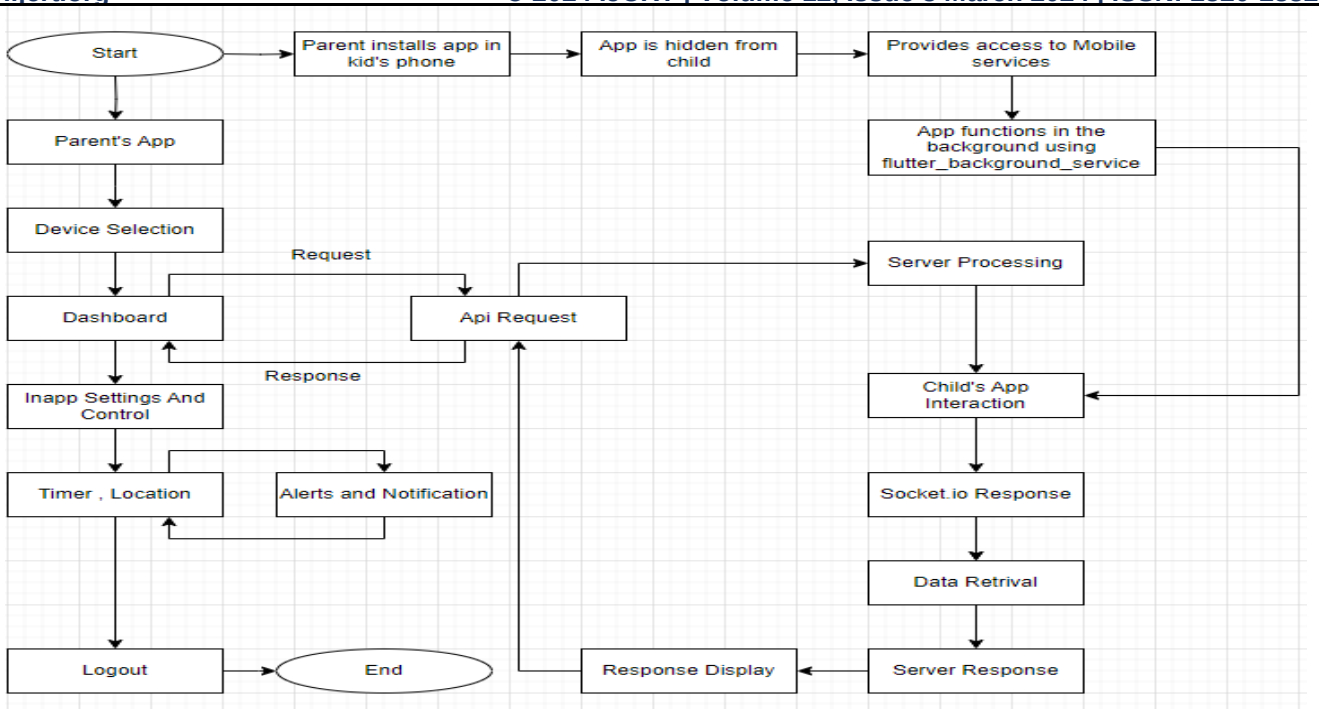


Fig. 1. Functionality flowchart

B. Algorithm

A client-server system for a Remote Access Trojan (RAT) project involves the client (the primary Flutter app), the server (Node.js), and the child (secondary Flutter app).

Primary(Parent) Flutter App:

- Initialization: Upon launch, the primary app initializes the communication channel by establishing a secure network connection with the Node.js server. This connection, often implemented using secure protocols such as HTTPS, serves as the foundation for secure and reliable data exchange between the parent device and the server. The initialization phase may involve implementing mechanisms for error handling and recovery.
- Authentication: The authentication process in the primary app is a critical step to ensure the security and legitimacy of user access. It involves assigning unique ids to the parent's device which will ensure security of data. Users are required to log in or provide relevant credentials, which are then securely transmitted to the Node.js server. The server, equipped with an authentication mechanism, verifies user identity. The authentication response, sent back to the primary app, includes information on the user's authentication status, allowing the app to proceed with authorized actions.
- API Request: Upon receiving API requests from the primary app, the server processes these requests in accordance with the defined API endpoints. This involves interacting with databases, external services, or other necessary resources to fulfill the requested actions. Error handling mechanisms are in place to address issues such as database connection failures or invalid requests, ensuring the reliability of API processing.
- Socket Event Handling: The server acts as a central hub for handling socket events originating from both the primary and secondary apps. It employs event-driven socket integration to efficiently route incoming events to their intended recipients. For instance, data from the secondary app is relayed to the primary app, and vice versa, ensuring seamless communication. The server's role in managing socket events enhances the responsiveness and real-time nature of the entire system.
- Data Transmission: the server facilitates data relay between the primary and secondary apps. When data needs to be transmitted from one app to the other, the server receives the data from the sending app and forwards it securely to the receiving app through a series of socket events. This relay mechanism guarantees the security, and efficient transmission of data between the parent and child devices.

Server (Node.js):

- Initialization: The Node.js server's initialization involves starting the server and configuring it to listen on a designated port. This process includes setting up the necessary server-side security measures. The server establishes secure connections with both the primary and secondary apps over the network, creating a reliant communication infrastructure.
- Authentication: Authentication requests from the primary app trigger a verification process on the server side. The server, equipped with secure authentication protocols, validates user credentials by comparing them against stored information in a secure database. Successful authentication results in the generation of authentication tokens or identifiers, which are sent back to the primary app.
- API Processing: Upon receiving API requests from the primary app, the server processes the requests. It may interact with a database or perform other necessary actions to fulfill the API request.
- Socket Event Handling: The server listens for socket events from both the primary and secondary apps. It routes socket events to the appropriate recipient (e.g., forwarding secondary app data to the primary app).
- Data Relay: If data needs to be transmitted between the primary and secondary apps, the server acts as a relay. It receives data from one app and forwards it to the other app via socket events.

Secondary(Child) Flutter App:

- Initialization: The secondary app initializes by establishing a socket connection with the Node.js server. This involves implementing secure socket communication protocols to ensure efficient data exchange between the child device and the server. The initialization phase may also include device registration processes to uniquely identify and authorize the secondary app. An authentication id needs to be passed which will be matched to that of the id passed during the initialization phase of the app on the parent device. This ensures security of data.
- Socket Connection: The secondary app establishes a socket connection with the Node.js server. This persistent connection allows the parent's app to seamlessly request data for different child activities like sms, call logs, location and app-usage. It involves error handling such as when socket connections fail between the server and the child app then the parent will be notified by appropriate error message.
- Data Transmission: The secondary app can send data or events to the primary app via socket communication. It constructs and sends socket events to the server, which forwards them to the primary app.
- Socket Event Handling: The secondary app listens for socket events sent by the Node.js server. It processes incoming socket data or events and may update the user interface or perform actions. Based on the socket ids provided by the websocket wrapper library the parent can get access to the data of the child device's activities. This ensures security of data among multiple users in the system.

SMS Spam Detection:

- Dataset: The dataset contained 5572 messages from different sources, out of which about 750 messages were spam and the remaining 86% messages were ham as seen in Fig. 2. The dataset has two columns - Message, Type(spam or ham).

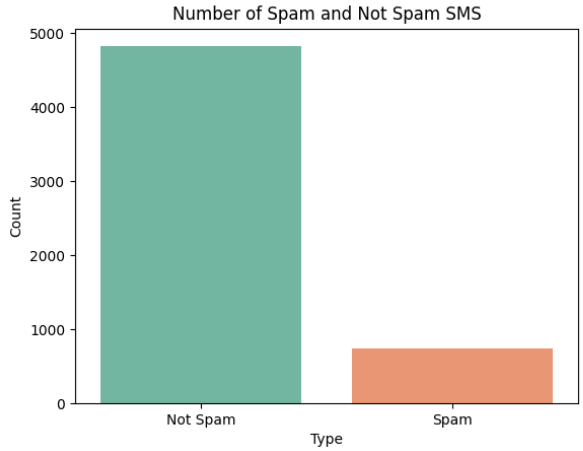


Fig. 2. Distribution of SMS

- Multiple classification models like Logistic Regression, Random Forest, Decision Tree and Naive Bayes were trained and compared to find the model with the highest accuracy Fig. 3.

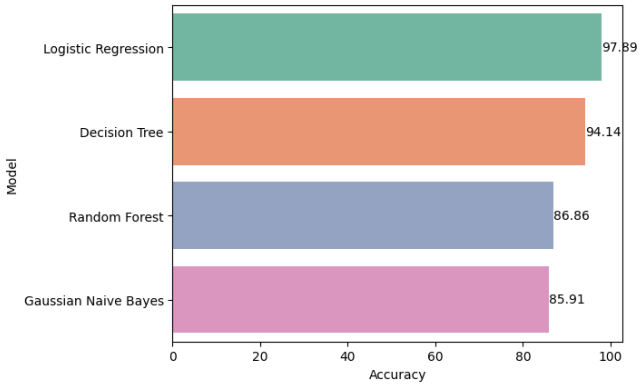


Fig. 3. Spam Detection Model Accuracy

This algorithm represents a basic flow of how the client-server system for a RAT application might operate. In a real-world scenario, there would be additional considerations for privacy, security, error handling, and other functionalities personalized as per one's requirements.

V. RESULTS

The software result of the Remote Access Tool is a comprehensive and user-centric mobile application developed using the Flutter framework. This application serves as a powerful solution for parents and authorized administrators to remotely monitor a host device's activities, ensuring safety and responsible digital behavior.

A. Realtime Monitoring

The Application enables real-time access to crucial data on the host device, including SMS messages, call logs, contacts, and real-time location information. This feature ensures parents can stay informed about their child's digital interactions and physical whereabouts.

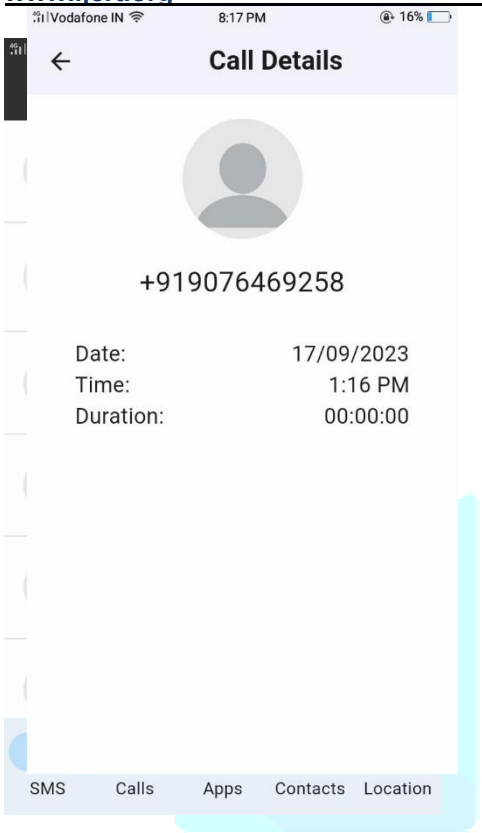
B. Support For Different Generations Of Devices

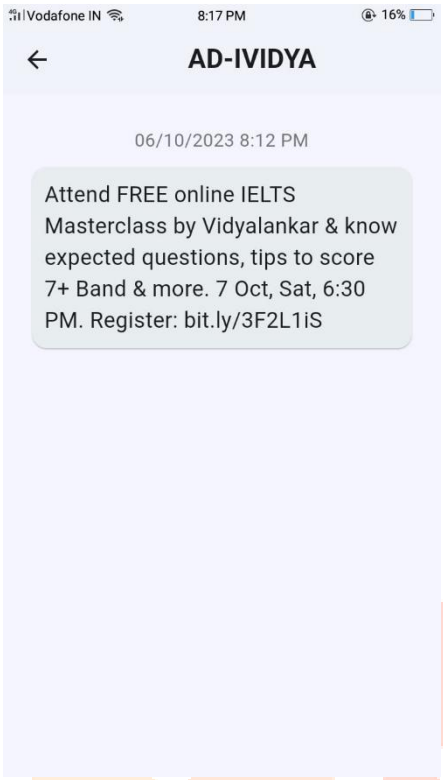
Tested the compatibility of the application by downloading, installing and running all the features on multiple devices as follows –

No	Device Name	Android Version	RAM (GB)	Installation	Function Working
1	Oppo A71	7	4	No Errors	All
2	Virtual Device	8	4	No Errors	All
3	Redmi Y2	9	3	No Errors	All
4	Redmi 9 Prime	10	4	No Errors	All
5	Vivo Y20	10	4	No Errors	All
6	Realme 5i	11	4	No Errors	All
7	Virtual Device	11	4	No Errors	All
8	Vivo V20	11	8	No Errors	All
9	Samsung Galaxy M51	12	6	No Errors	All
10	Realme 8 5G	13	8	No Errors	All
11	Moto Edge 40	13	4	No Errors	All

C. User Interfaces

User-friendliness is at the forefront of the software design, featuring intuitive and easy-to-navigate interfaces for both the host and remote monitoring apps, enhancing the user experience.





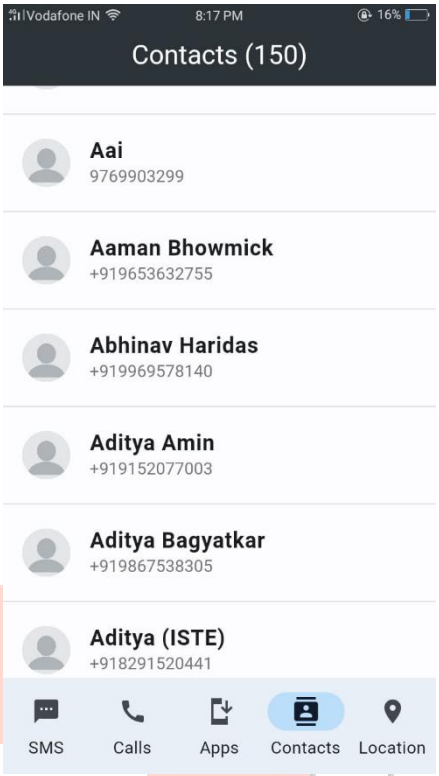
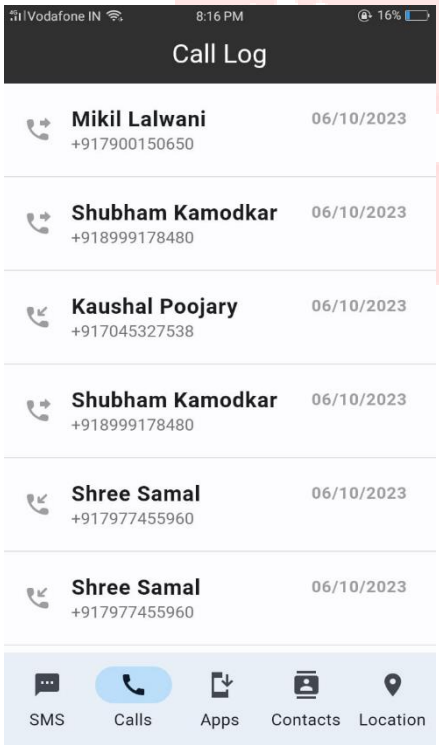


Fig. 4. SMS as seen on Primary device. Fig. 5. Content of SMS shown on Primary device.
Fig. 6. Call Log Seen on Primary device.



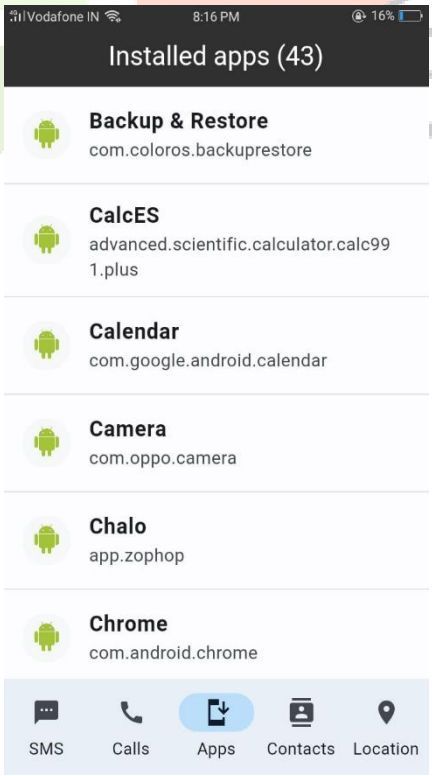


Fig. 7. Call Details as seen on Primary device. Fig. 8. App Installed as seen on Primary device.
Fig. 9. Contacts as seen on Primary device.

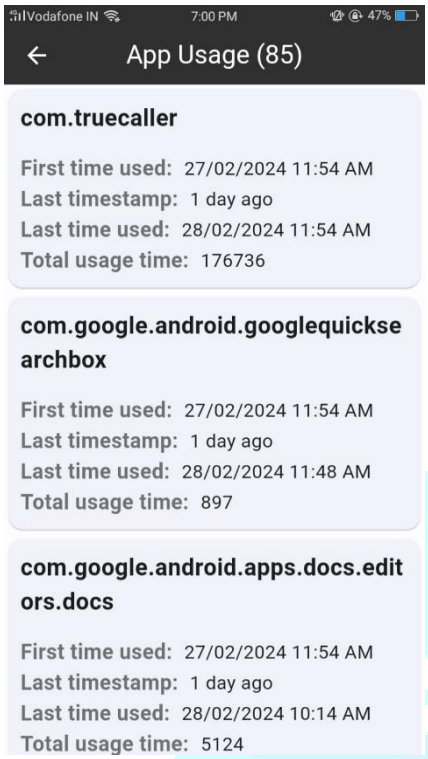


Fig. 10. Child device's app usage on parent device.

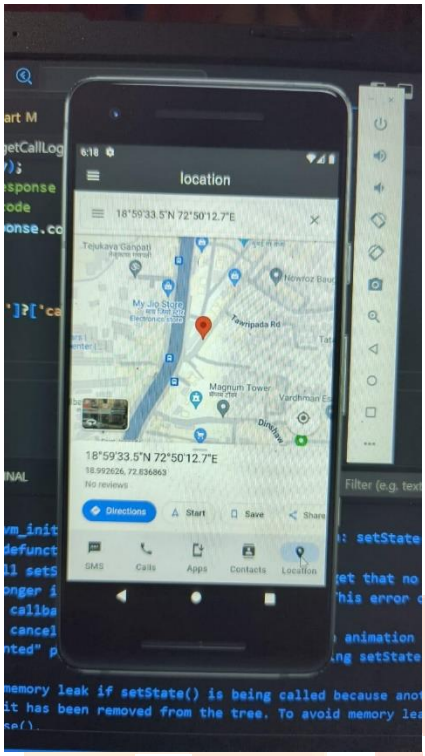


Fig. 11. Live Location as seen on Primary device.

Fig. 12. Spam SMS alert on Primary Device

D. Evaluation

Testcase	**State of child app**	**State of parent app**	**Result on child app**	**Result on parent app**
1.	Child connected	Parent connected	Child activities are available to be monitored	Parent notification received or sequence of activity done
2.	Child connected but temporarily out of range	Parent connected	Child app will send an event to notify that proximity is crossed.	Parent app will receive notification that the child has crossed the notification with live location.
3.	Child connected but permanently out of range	Parent connected	Child app will send an event to notify that proximity is crossed.	Parent app will receive notification that the child has crossed the notification with live location.
4.	Child mobile off	Parent connected	No activity will be monitored	Parents will receive an error message indicating the child is offline.

5.	Child connected	Parent not connected	Child activities will remain available to be monitored	Parents need to connect to monitor child activity.
6.	Child connected	Parent temporarily not connected	Child activities will remain available to be monitored	Parents will receive notification for the child activity once connected.

E. Future Scope

In this project, privacy and security of data is of utmost importance. Striking the perfect balance between privacy and the safety of the client is imperative. To further enhance this system, a model can be developed which will provide a summary of the user's emotion based on the user's search history. This model will not be giving any direct search queries which will ensure privacy while providing the caregivers a brief overview of the mental state of their ward.

An analytics page can be developed to show the user's statistics in a graphical manner. This will help the caregivers to analyze trends, take actions and monitor the progress over a period of time.

Past data can be saved on cloud, so that the primary device can access it even if the secondary device is offline. This will also help to create a backup copy of the data in case the data were deleted from the secondary device.

VI. CONCLUSIONS

The Remote Access Tool project revolves around the development of a versatile mobile application using the Flutter framework. This tool enables remote monitoring of a host device from a separate remote device, primarily designed to empower parents to supervise their child's mobile phone usage. The system's workflow involves the integration of Flutter and relevant packages for accessing SMS messages, call logs, contacts, and real-time location data. It ensures secure data transmission and user authentication, offering intuitive and user-friendly interfaces. With the integration of SMS spam detection, the parent can monitor such text messages and further take action after receiving an alert notification. Also the child's live location will always be accessible to the parent on his device which will be updated periodically. The parent can also set a range or radius and if the child moves out of the permitted area then the parent will be notified. Beyond enhancing child safety and promoting responsible digital behavior, this tool's adaptability allows it to serve various other scenarios, such as elderly care or employer device monitoring, making a positive impact on digital safety and responsibility across different segments of society.

VII. ACKNOWLEDGEMENT

The project report on "Remote Access Tool" is the outcome of the guidance, moral support and devotion bestowed on our group. For this we acknowledge and express our profound sense of gratitude to everybody who has been the source of inspiration throughout project preparation. First and foremost we offer our sincere phrases of thanks and innate humility to H.O.D Dr.(Mrs.)Shalu Chopra , Project guide Dr. Manoj Sabnis and other professors who provided valuable inputs and the consistent guidance and support provided by them. We thank Vivekanand Education Society's Institute of Technology for providing such a stimulating atmosphere and conducive work environment.

REFERENCES

[1] S. P. Walsh, K. M. White, and R. M. Young, "Over-connected? A qualitative exploration of the relationship between Australian youth and their mobile phones," Journal of Adolescence, vol. 31, no. 1, pp. 77–92, Feb. 2008.

[2] "Teens and Internet, Device Access Fact Sheet 2024 | Pew Research Center's Internet & American Life Project." [Online]. Available: https://www.pewresearch.org/internet/fact-sheet/teens-and-internet-device-access-fact-sheet/ [Accessed: 5-Jan-2024].

[3] Yuanyi Chen, "Application System of Remote-Access Computer of Android Mobile Phone", Jiangxi Teachers College, China, 2021.

[4] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," IEEE Communications Magazine, vol. 48, no. 9, pp. 140–150, 2010.

[5] Y. Lee and S.-B. Cho, "Extracting Meaningful Contexts from Mobile Life Log," in Intelligent Data Engineering and Automated Learning-IDEAL 2007, H. Yin, P. Tino, E. Corchado, W. Byrne, and X. Yao, Eds. Springer Berlin Heidelberg, 2007, pp. 750–759.

[6] Selwyn, N. and Aagaard, J. (2021), Banning mobile phones from classrooms—An opportunity to advance understandings of technology addiction, distraction and cyberbullying. Br. J. Educ. Technol., 52: 8-19. https://doi.org/10.1111/bjet.12943

[7] K.S. Kuppusamy, Leena Mary Francis, G. Aghila, "RePort: A Model for Remote Parental Control System Using Smartphones", Department of Com-puter Science, School of Engineering and Technology, Pondicherry University,Pondicherry, India, 2013.

[8] Álvaro Feal, Paolo Calciati, Narseo Vallina-Rodriguez, Carmela Troncoso, and Alessandra GorlaAngel or Devil? A Privacy Study of MobileParental Control Apps,2020.

[9] FamiSafe (2023), Shenzhen Wondershare Software Co., LTD. Accessed: Jan. 15, 2024. [Online]. Available: https://play.google.com/store/apps/details?id=com.wondershare.famisafe&hl=en_IN&gl=US

[10]KidsGuard(2023), ClevGuard HK. Accessed: Feb. 1, 2024. [Online]. Available: https://play.google.com/store/apps/details?id=com.clevguard.kidsguard.parent&hl=en_IN&gl=US

[11] Houshmand Shirani-Mehr, SMS Spam Detection using Machine Learning Approach.