# Covid-19 digital Contact-tracing: a doorway to well-being or a backdoor to security vulnerabilities?

Nishit Patel*, David Cancel*, Moitrayee Chatterjee*, Md Shahinoor Rahman†
*Department of Computer Science*, Department of Earth and Environmental Sciences†*
*New Jersey City University*
{npatel10, dcancel3, mchatterjee, mrahman1}@njcu.edu

***Abstract*—Digital Contact-tracing through mobile applications require gathering of location and other personal information of an individual by the government or private organizations and became an essential solution for moderating the pandemic and slackening lockdown measures. However, the moral and legal boundaries for such privacy-sensitive information reconnaissance procedure and the ambiguity in the security measures of such technologies has gained controversial reputation.**

**In this work, we performed static profiling of 10 different Android Contact-tracing applications, developed by the health departments of 10 different states within the United States and studied possible security threats posed by them. To the best of our knowledge, our work is the first to heuristically analyze the users′ attitude towards these applications to understand the user-perceived contribution of these apps towards their well-being. We collected user feedback for each of the apps and trained a logistic regression classifier on cleaned, pre-processed and vectorized texts to identify positive or negative outlook towards these apps. Using the confusion matrix, our predictive model showed up to $85\%$ accuracy, $94\%$ precision, $93\%$ recall and $83\%$ f1 score. in predicting the sentiments. The sentiment prediction shows, users in some states did find the apps to be helpful where some other states found them wasteful. Whereas, our static analysis shows none of the apps are malicious themselves but all of them request permission that can be abused to gain escalated privileges.**

***Index Terms*—Security, Usability, Covid-19, Contact-Tracing, User Sentiment.**

## I. Introduction

Contact-tracing is a strategy to distinguish people who may have come in close contact with an infected individual while that individual was the carrier of the virus. Contact-tracing has been utilized as an effective mechanism to control and screen previous outbreaks like HIV, Ebola, and measles. The traditional method of Contact-tracing is manual [12]. Apparently straightforward, Contact-tracing is the most common way of recognizing all individuals that a COVID-19 patient has interacted over a period of past fourteen days. But in reality, this task can be hard to perform manually because: (1) it requires a sizeable amount of professional to perform the tracing mechanisms; and (2) it can not recognize people that are not related to the infected individual.

Moreover, manual Contact-tracing puts the professionals at greater risk of contacting the disease themselves as Covid-19 is highly transmissible. Since the outbreak, the Covid-19 infection has grown exponentially[1]. Hence, governments and private health organizations have lean towards more scalable, time and cost efficient and most importantly socially distant, technologically based Contact-tracing solutions [22].

The wide adoption of cell phones has given a chance to create applications that leverage short-range correspondence between the mobile phones to detect their spatial closeness. The low-power Bluetooth packet exchanges between phones translates into vicinity measures between people whose mobile phones are running the Contact-tracing application. And whenever an individual is tested positive for the Covid-19, a notification can be sent to all other individuals who came in close contact with the infected individual in recent past [13].

The collection and utilization of digital information is introduced as a vital cure by the government and private administrations across nations. Moreover organizations and health specialists and scientists from diverse fields have identified the expansive scope of data-driven analysis that can be fulfilled by gathering, breaking down, and sharing information from various digital sources. These sources incorporate information from phone towers, cell phone applications, Bluetooth associations, CCTV footage, social web, financial transactions, wearable devices to name a few. In conjunction, Apple Inc and Google Inc introduced their DP-3T and TCN protocols-inspired collaborative *Google Apple Exposure Notification (GAEN)* Contact-tracing technology [5], [8].

With the availability of enormous amount of digital data on individuals around the globe, and the advent of big data technologies the hope of finding data driven solutions is promising. However, this promise comes with two major concerns: (1) the precision, and quality of information that change enormously across the various sources, and the inter-operability issues and security risks. (2) legal and ethical boundaries in information handling, and the inherent security threats introduced through these new Contact-tracing technologies. Hence, it is imperative to understand the adoption and effectiveness of these applications along with the security threats posed by them.

The contribution of this project is two-fold:

(1) Technical: the work presented in this paper scrutinizes randomly selected Covid-19 Contact-tracing apps for Android platform. We utilized Android reverse engineering tool Androguard [1] for finding:

(a) Sensitive permissions, API call involving end-user privacy.

(b) Study app configuration for vulnerabilities and weaknesses through Common Weakness Enumerations (CWEs) [2], and

[1] https://www.seti.org/coronavirus-and-exponential-growth-updated-4-20-2020

Common Vulnerabilities and Exposures (CVEs) [6].
And, to the best of our knowledge, our work is the first to analyze the usability and effectiveness study of the Contact-tracing apps from a user perspective.

(2) Fostering Undergraduate Research through student and faculty mentor collaboration: This current work was done as part of New Jersey City University′s undergraduate (UG) summer research internship to systematically improve cognitive and applied learning for UGs and enhance their overall career and educational experience, by working closely with faculty mentors in their distinguished field.

The rest of the paper is organized as follows: we discuss the motivation for this work in the Section II. Then we provided the overview of our approach in the Section III. We iterated the existing technologically based Contact-tracing for Covid-19 in Section IV. The Section V provides insights about our findings. Then we concluded the paper in Section VI.

## II. Motivation: STEM Research Internships in the time of a global pandemic

New Jersey City University is a minority serving, public institution in a urban setting. Its STEM (Science, Technology, Engineering and Math) Research Internship opportunities, over the summer semesters present the undergraduate students with research engagements and work-based learning experiences [7]. According to U.S. Bureau Of Labor Statistics, STEM occupations remains economically more competitive than non-STEM occupations of the United States, with median annual wage of $\$89,780$ as for 2020 [4]. During the internship, the students get to expand their knowledge and skills in a chosen area by working closely with a faculty mentor. Moreover, the internship opportunities provides the students to build stronger critical thinking and analytical skills, which help them to advance their academics and career goals beyond the classroom.
With the ongoing Covid-19 pandemic of a massive proportion when increasing reliance on technology for education, work, living and staying connected while in isolation, it only deemed fitting to our team of UG interns and faculty mentors to study how the technology is aiding our lives. Mobile based Contact-tracing apps have been advocated all over the world for instantaneous exposure notifications. These applications has also gained a tainted reputation for encroaching user privacy and posing potential security threats. While, the privacy and security concerns of the country-specific Contact-tracing apps have been surveyed extensively [9], [10], [21], to the best of our knowledge our work is the first on analyzing the usability of the apps from a user′s perspective. Also, instead country specific apps, we randomly selected 10 out of 28 Android application developed by the various state health departments within the United States for our study. We first static profiled these applications to analyze the possible privacy invasion and security concerns first hand. Then, we crawled the Google Play Store to gather textual feedback of the users who downloaded and used these apps. We applied predictive analysis and modeling on the textual data. We classified the user feedback to analyze if they felt strongly or otherwise about the contribution of these apps towards their well being during the pandemic.

## III. Methodology

We present the details of our experimental set up and experimental overview in this section.

### A. Static App Profiling for Security Analysis

We setup our experimental environment for static profiling the Android apps on the Oracle VM VirtualBox[2] on MacOs and Windows systems. Then we installed Kali Linux[3] on our VirtualBox. This Debian-based Linux distribution comes with tools for performing penetration testing and security analysis. We installed Python based tool Androguard [1] on our Kali Linux machines. Androguard has been the “Swiss army knife” for Android forensics, malware and goodware detection [11], [27]. We downloaded Android Package Kit (apk) files for the chosen apps on our Kali Linux. Then we decompiled each of the apk file using Androlyze utility of Androguard to check for the behavior of the apps [3], that could pose potential security threats:

- Services in the app: Through the “service” component an app can serve as the gateway for another app to acquire other functionalities. Services can keep running even if the user is not interacting with the app that started it.
- Activities invoked by the applications: Activities not only serve as the first point of interaction between the application and the user, but the activities within one application can potentially invoke another activities in another applications.
- Files in the app: Applications need access to internal storage to store the data that are necessary for the app to run. However, it was imperative for our study to find out if any of the apps were having access to other shareable files outside the app′s specific directories.
- Permissions requested by the app and potentially dangerous permissions: Through various permission requests (for example: install time- permission, run time permission, special permission and so on) application can have access to privacy sensitive information present at the hardware level or user information or even information present in another application, making it important to scrutinize for elevated privileges provided to an app.

### B. Usability Study: Sentiment Analysis

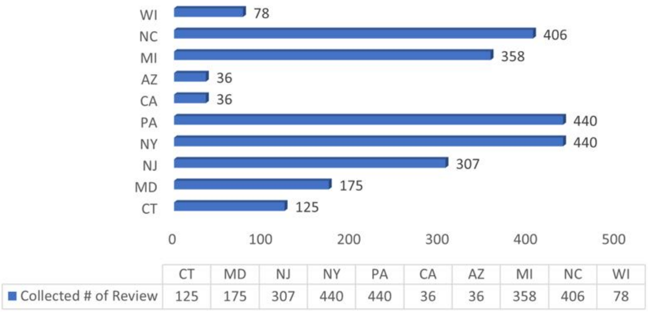
The ISO 9241-11 details the structure for understanding the usability and applying it to circumstances where individuals utilize intelligent and automated products. IBM′s Computer System Usability Questionnaire (CSUQ) [20] outlines the subjective usability measurements. However, most of the subjective and objective usability studies require extensive user opinion on various factors of the product to achieve the desired level of effectiveness and efficiency in a specified context.
In the defection of formal method we heuristically performed

[2] https://www.virtualbox.org/
[3] https://www.kali.org/

the usability study [19], [23] for the 10 Android apps. We scraped user feedback for each app from the Google Playstore[4] and performed data driven predictive analysis to classify overall user sentiment about each of the 10 app.

For experimental setup, we used browser based Python notebook Colaboratory[5] to develop necessary code for performing these steps. And, the following steps describe our approach for heuristic based usability study:

- *Data Collection:* We developed the Playstore review scraping algorithm using methods from *BeautifulSoup* library of Python. The Fig. 1 shows the number of user review we collected per state-app. Due to space constraint we used the official abbreviations[6] for each state, instead of the full app names.



| | CT | MD | NJ | NY | PA | CA | AZ | MI | NC | WI |
|---|---|---|---|---|---|---|---|---|---|---|
| ■ Collected # of Review | 125 | 175 | 307 | 440 | 440 | 36 | 36 | 358 | 406 | 78 |

Fig. 1. Number of Reviews Scraped per State,

- *Data Cleaning and Pre-processing:* The review data scraped from the Playstore were much unorganized. So, we utilized various functions from Python′s Natural Language processing *NLTK (Natural Language Toolkit)* library to remove url, emoticons and other non-alphabets including punctuation and converting all letters to lower case.
- *Text Classification:* Logistic regression algorithm and its variants have been proven efficient in text classification [15], [18]. We used one-hot encoding to vectorize the cleaned and pre-processed data. The vectorized data is then split into train (80%) and test (20%) to train the logistic regression classifier. Based on the textual opinions from the users, the classifier predicted binary outcome of user sentiment as $-1$ and 1, that we further translated as negative and positive sentiment, respectively.
- *Evaluate Model Performance:* We used confusion matrix to analyze the accuracy, precision, recall and f1 score of our classifier.

## IV. Related Work

Most of the recent studies on COVID-19 contact-tracing can be broadly categorized into two groups: comparison of available apps based on their features and discussion on privacy policy guidelines [14], [17], [24]–[26]. Several studies evaluated different technologies, performances, user adaptations, and most importantly privacy as well as data security issues [24], [26]. Trivedi and Vasisht compared privacy concerns of five different contact tracing technologies: Bluetooth, GPS, acoustic, wifi logs, and non-mainstream advanced approaches [26]. They found tracing accuracy is mostly inversely related to privacy concerns, for instance acoustic and advanced approaches have high proximity accuracy while having significant privacy concerns, on the other hand, bluetooth and wifi logs have low accuracy but preserve privacy. Another study by Redmiles [24], explored the trade-offs and user concerns for the adaptation of a COVID-19 contact-tracing app through technology benefits, solution accuracy, privacy considerations, and mobile-related costs. This study concluded that users have several privacy and surveillance concerns including the personal information is being leaked to cyber attackers and disclosure of their location trajectories. Sharon [25] conducted an interesting study on trust of contact-tracing app users and found that people have more reliance on data security and privacy provided by private tech giants, compared to demographic government. Therefore, tech cooperates are encroaching into spheres of social life, which raise various risk that often is not included in the discussion of privacy harm. Gasser et al. discussed the legal boundaries and ethical considerations of data protection and privacy using COVID-19 contact-tracing apps typology [14]. They concluded that digital tools for measuring relative spatial proximity is less privacy invasive than personal contact tracing.

## V. Results

This sections discusses the results and findings of (A) static profiling the applications, (B) results of usability study (C) evaluation of the model trained for usability study.

### A. Static Profiling: Security vulnerabilities in Contact-tracing apps

Using Androguard tool we gathered the permissions requested by the apps. The Table I lists down the common permissions requested by all the 10 apps we static profiled. We categorized these permissions into hardware level and software level.

At the time of study, we searched for vulnerabilities related to each of the permissions in the National Vulnerability Database [6]. The GAEN [5], [8] and Android Bluetooth permissions were associated to severe vulnerabilities. *GAEN* permission′s associated vulnerability CVE−2021−31815 describes that attackers can leverage this permission within the app to gain sensitive information about the phone′s user. The related weakness CWE−319 facilitates transmission of sensitive information in clear text.

*BLUETOOTH* permission′s associated vulnerabilities CVE−2020−0022 enables remote code execution whereas CVE−2021−25430 lets an attacker to gain improper access to the device. Related weakness CWE−287 can not authenticate a user from an attacker apart and CWE−787 lets a code

[4] https://play.google.com/store

[5] https://colab.research.google.com/

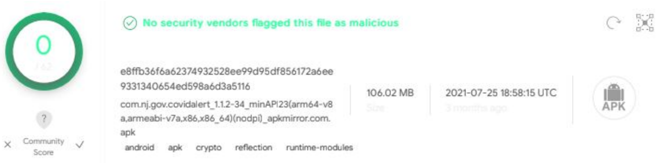
[6] https://about.usps.com/who-we-are/postal-history/state-abbreviations.htm

| Level | App permissions | Related Data Access | Possible Attacks |
|---|---|---|---|
| Hardware Level | INTERNET<br>BLUETOOTH<br>RECEIVE_BOOT_COMPLETED<br>VIBRATE (NJ and NY apps*) | Device Level Information:<br>OS Version, Device,<br>Model, API level,<br>Firmware Version. | Denial Of Service,<br>Injection attacks,<br>Man In The Middle<br>attack |
| Software Level | FOREGROUND_SERVICE<br>ACCESS_NETWORK_STATE<br>WAKE_LOCK<br>BIND_GET_INSTALL_REFERRER_SERVICE (AZ app*) | User Information:<br>Last Name, First Name,<br>Email, ID, Profile<br>Photo. | Enumeration attacks,<br>Tracking and<br>de−anonymization<br>attacks |

TABLE I
SUMMARY OF APPS PERMISSIONS.
(*AT THE TIME OF THE STUDY.)

execution using out−of−bound memory location.
*VIBRATE* permission was requested by on NJ (Covid Alert NJ) and NY (Covid Alert NY) apps and *BIND_GET_INSTALL_REFERRER_SERVICE* permission was requested in the AZ (Covid Watch AZ) app. Based on the CVE and CWE descriptions, and the work of Gvili [16], we listed down the possible attacks related to these permissions in the thrird column of Table I.

However, based on the services opened, activities invoked and the related file information our team did not recognize any maliciousness of the apps. To cross validate our decision, we uploaded individual apk files for the apps to VirusTotal[7] to scan for potential maliciousness. As shown in Fig. 2 none of the 10 apk were flagged found malicious by the security vendors of VirusTotal.



No security vendors flagged this file as malicious

e8ffb36f6a62374932528ee99d95df856172a6ee9331340654ed598a6d3a5116
com.nj.gov.covidalert_1.1.2-34_minAPI23(arm64-v8a,armeabi-v7a,x86,x86_64)(nodpi)_apkmirror.com.apk
android apk crypto reflection runtime-modules

106.02 MB
Size

2021-07-25 18:58:15 UTC
3 months ago

Fig. 2. Snapshot of Covid Alert NJ app using Virustotal scan.

## B. Usability Study

The review data set was split into $80\%$ and $20\%$ to train the logistic regression classifier. A value of $0.5$ was used as the classification threshold. Sentiment predicted below the threshold is labeled as negative and equal or above is labeled as positive. After training the classifier we tested it of each of the state′s app reviews and we visualized the number of positive and negative sentiments for each of the states in Fig. 3. The blue bar represents the number of negative reviews and amber bar represents the number of positive reviews. As, presented in Fig. 3, there are more satisfied users of CT, PA, NJ, NY, NC, WI apps whereas, majority of the users of MD, CA, MI, AZ apps are dissatisfied.

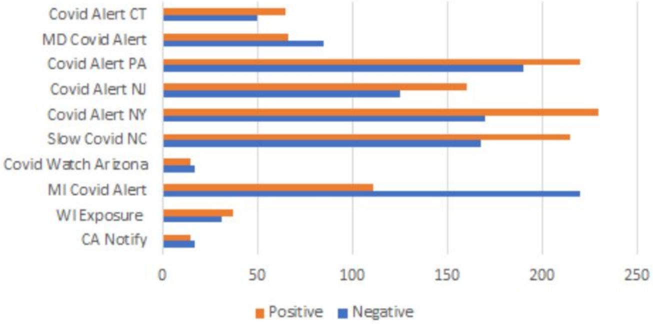


Fig. 3. Predicted Sentiment

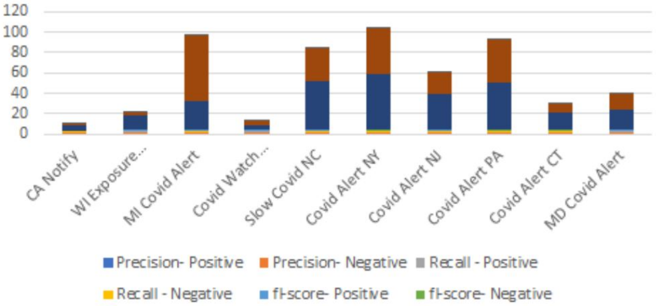


Fig. 4. Model Performance

## C. Model Evaluation

We leveraged a confusion matrix as the performance summary of our logistic regression classifier. We have visualized the confusion matrix of our predictive model using a bar graph and presented in the Fig. 4. The confusion matrix showed the model performed the best for the MD app (MD Covid Alert) by predicting user sentiment with $85\%$ accuracy, $94\%$ precision, $93\%$ recall and $83\%$ f1 score.

## VI. CONCLUSION AND FUTURE DIRECTION

Contact-tracing mechanisms measure the spatial proximity among individuals to determine their exposure. Contact-tracing, can recognize the point when individuals are presented to a person that is positive for Covid-19. As part of the effort to reduce the spreading of the virus infection, governments and health organizations have advocated digital contact-tracing mechanisms. Within the United States, 28 states′ health department developed such Contact-tracing apps.
The work presented in this paper performed static analysis of 10 Android apps developed by 10 states in USA, using Androguard tools. The result from the static analysis were further studied with regards to CVE and CWE enumeration to understand the possible security threats posed by these apps.

[7] https://www.virustotal.com/gui/home/upload

Furthermore we collected user feedback from the Playstore for these 10 apps and used logistic regression to predict user sentiments towards the apps.

The static analysis of the app pointed that while the apps are not malicious themselves, the permissions requested by those apps are the gateway for attackers and other malicious apps to gain escalated privileges by exploiting the vulnerabilities and launch various cyber attacks.

The logistic regression based predictive model classified the user feedback into binary classes: positive and negative. The classification results show that the users of CT, PA, NJ, NY, NC, WI Contact-Tracing apps have more positive sentiment towards the apps whereas the users of MD, CA, MI, AZ feel more negatively about the apps.

Apart from our technical findings, this project was a success in involving and motivating our UG students through Summer Research Internship. This project introduced the participating students to the concepts of Android app forensics, basic natural language processing and machine learning algorithms.

The usability study was performed on unlabeled data and only logistic regression was used for binary classification for the sentiment prediction. However, current implementation did not consider neutral sentiment. Performance of other machine learning algorithms like KNN or Naive Bayes for our dataset need to be studied. Furthermore, this work can be extended to study the usability of the apps subjectively and objectively.

## Acknowledgment

We express our gratitude to Dr. John C. Grew, Professor of Biology, and Ms. Kristina Harb, Title V Program Assistant at New Jersey City University for their initiative and help in organizing the teams of students and faculty mentors.

This work is funded by:

(1) US Education Department Title III Part F HSISTEM Grant number P031C160155 “Proyecto STEM: Evidence-Based Approaches to STEM Enrollment, Retention and Graduation at an Urban Public Hispanic-Serving Institution”.

(2) US Education Department Title V DHSI Grant number P031S200124 “On PACE for STEM Success: Personal, Academic & Career Enhancement at an Urban Public Hispanic-Serving Institution”.

## References

[1] Android reverse engineering tool Androguard. https://github.com/androguard/androguard, 2021. [Online; accessed 29-Oct-2021].

[2] COMMON WEAKNESS ENUMERATION. https://cwe.mitre.org/index.html, 2021. [Online; accessed 29-Oct-2021].

[3] Documentation for Android Developers. https://developer.android.com/docs, 2021. [Online; accessed 29-Oct-2021].

[4] Employment in STEM occupations. https://www.bls.gov/emp/tables/stem-employment.htm, 2021. [Online; accessed 29-Oct-2021].

[5] Exposure Notifications: Helping fight COVID-19 - Google. https://www.google.com/covid19/exposurenotifications/, 2021. [Online; accessed 29-Oct-2021].

[6] NATIONAL VULNERABILITY DATABASE. https://nvd.nist.gov/, 2021. [Online; accessed 29-Oct-2021].

[7] NJCU Grant-Funded Opportunities and Resources. https://www.njcu.edu/academics/schools-colleges/william-j-maxwell-college-arts-sciences/departments/biology/grant-funded-opportunities-and-resources, 2021. [Online; accessed 29-Oct-2021].

[8] Privacy-Preserving Contact Tracing. https://covid19.apple.com/contacttracing, 2021. [Online; accessed 29-Oct-2021].

[9] Nadeem Ahmed, Regio A Michelin, Wanli Xue, Sushmita Ruj, Robert Malaney, Salil S Kanhere, Aruna Seneviratne, Wen Hu, Helge Janicke, and Sanjay K Jha. A survey of covid-19 contact tracing apps. *IEEE access*, 8:134577–134601, 2020.

[10] Samuel Altmann, Luke Milsom, Hannah Zillessen, Raffaele Blasone, Frederic Gerdon, Ruben Bach, Frauke Kreuter, Daniele Nosenzo, Séverine Toussaert, Johannes Abeler, et al. Acceptability of app-based contact tracing for covid-19: Cross-country survey study. *JMIR mHealth and uHealth*, 8(8):e19857, 2020.

[11] Nguyen Tan Cam, Toan Nguyen, Khanh Nguyen, Tuan Nguyen, and Van-Hau Pham. Detect malware in android firmware based on distributed network environment. In *2019 IEEE 19th International Conference on Communication Technology (ICCT)*, pages 1566–1570. IEEE, 2019.

[12] Ken TD Eames and Matt J Keeling. Contact tracing and disease control. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270(1533):2565–2571, 2003.

[13] Luca Ferretti, Chris Wymant, Michelle Kendall, Lele Zhao, Anel Nurtay, Lucie Abeler-Dörner, Michael Parker, David Bonsall, and Christophe Fraser. Quantifying sars-cov-2 transmission suggests epidemic control with digital contact tracing. *Science*, 368(6491), 2020.

[14] Urs Gasser, Marcello Ienca, James Scheibner, Joanna Sleigh, and Effy Vayena. Digital tools against covid-19: taxonomy, ethical challenges, and navigation aid. *The Lancet Digital Health*, 2020.

[15] Alexander Genkin, David D Lewis, and David Madigan. Large-scale bayesian logistic regression for text categorization. *technometrics*, 49(3):291–304, 2007.

[16] Yaron Gvili. Security analysis of the covid-19 contact tracing specifications by apple inc. and google inc. *IACR Cryptol. ePrint Arch.*, 2020:428, 2020.

[17] Marcello Ienca and Effy Vayena. On the responsible use of digital data to tackle the covid-19 pandemic. *Nature medicine*, 26(4):463–464, 2020.

[18] Georgiana Ifrim, Gökhan Bakir, and Gerhard Weikum. Fast logistic regression for text categorization with variable-length n-grams. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 354–362, 2008.

[19] Laurie Kantner and Stephanie Rosenbaum. Usability studies of www sites: Heuristic evaluation vs. laboratory testing. In *Proceedings of the 15th annual international conference on Computer documentation*, pages 153–160, 1997.

[20] James R Lewis. Ibm computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1):57–78, 1995.

[21] Ichiro Nakamoto, Ming Jiang, Jilin Zhang, Weiqing Zhuang, Yan Guo, Ming-Hui Jin, Yi Huang, and Kuotai Tang. Evaluation of the design and implementation of a peer-to-peer covid-19 contact tracing mobile app (cocoa) in japan. *JMIR mHealth and uHealth*, 8(12):e22098, 2020.

[22] Cong T Nguyen, Yuris Mulya Saputra, Nguyen Van Huynh, Ngoc-Tan Nguyen, Tran Viet Khoa, Bui Minh Tuan, Diep N Nguyen, Dinh Thai Hoang, Thang X Vu, Eryk Dutkiewicz, et al. A comprehensive survey of enabling and emerging technologies for social distancing—part i: Fundamentals and enabling technologies. *IEEE Access*, 8:153479–153507, 2020.

[23] David Pinelle, Nelson Wong, and Tadeusz Stach. Heuristic evaluation for games: usability principles for video game design. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1453–1462, 2008.

[24] Elissa M Redmiles. User concerns 8 tradeoffs in technology-facilitated covid-19 response. *Digital Government: Research and Practice*, 2(1):1–12, 2020.

[25] Tamar Sharon. Blind-sided by privacy? digital contact tracing, the apple/google api and big tech’s newfound role as global health policy makers. *Ethics and Information Technology*, pages 1–13, 2020.

[26] Amee Trivedi and Deepak Vasisht. Digital contact tracing: technologies, shortcomings, and the path forward. *ACM SIGCOMM Computer Communication Review*, 50(4):75–81, 2020.

[27] Wei Wang, Ruoxi Sun, Minhui Xue, and Damith C Ranasinghe. An automated assessment of android clipboards. In *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, pages 1249–1251, 2020.