num compass

Conformity Check for Pandemic Health Apps

COMPASS Arbeitspaket 2_Best Practice & Guidance

In diesem Arbeitspaket sollen Best Practices definiert und Richtlinien formuliert werden, wie Apps je nach Fokus (Information, Beratung, Datenerhebung) innerhalb einer Pandemie-bekämpfung möglichst effektiv, effizient und forschungskompatibel eingesetzt werden können; dabei soll erarbeitet werden, welche Faktoren die Akzeptanz maßgeblich beeinflussen und daraus abgeleitet definiert werden, wie eine möglichst hohe Akzeptanz für Pandemieapps erreicht werden kann.

Version 1.0 _ 31.09.2021

netzwerk universitäts medizin

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Inhaltsverzeichnis

Ziele des Arbeitspaketes

In diesem Arbeitspaket sollen Best Practices definiert und Richtlinien formuliert werden, wie Apps je nach Fokus (Information, Beratung, Datenerhebung) innerhalb einer Pandemie-bekämpfung möglichst effektiv, effizient und forschungskompatibel eingesetzt werden können; dabei soll erarbeitet werden, welche Faktoren die Akzeptanz maßgeblich beeinflussen und daraus abgeleitet definiert werden, wie eine möglichst hohe Akzeptanz für Pandemieapps erreicht werden kann.

Dazu werden sukzessive die in AP 3-5 erarbeiteten Ergebnisse zu einer Gesamtbewertung zusammengeführt und die Empfehlungen entsprechend aktualisiert.

Um ein Hilfsangebot zu entwickeln, wie Netzwerkpartnern mit Bedarf an Pandemieapps bezüglich der Anforderungen und Möglichkeiten vorgehen können, sollen diese Richtlinien darüber hinaus operationalisiert werden, indem eine digitale Wissensbasis / FAQ erstellt wird.

Zuletzt sollen Beratungsstrukturen geschaffen werden, die in Interaktion mit identifizierten Partnern Auskunft zu den Guidelines und Angeboten aus diesem AP geben.

Abstract

Within the NUM-COMPASS project, effort is dedicated to the design of tools that evaluate an app's conformance with the protocols and standards set by the framework. In particular, the work package focuses on how a pandemic-related data model, e.g., German Corona Consensus Dataset (GECCO) can be integrated to a health app and which tests the integration implies. GECCO is an interoperable set of profiles for Fast Healthcare Interoperability Resource (FHIR) resources related to patient anamnesis in the context of COVID-19. Within NUM-COMPASS, tasks include: (1) To provide a framework, which checks whether a pandemic-related application, which produces FHIR profiles, isin conformity with the GECCO data model. (2) To check whether unnecessary sensible data is retrieved from the user and sent to third party clients. In this document, we present our preliminary analysis and requirement specification of this conformity checking task.

Introduction

The COVID-19 pandemic has outlined the necessity for the development of unified digital tools to facilitate pandemic research. The NUM-COMPASS project aims at designing a unified pipeline for app-based study designs. A crucial aspect of the development of a unified app pipeline for pandemic research is conformity testing. In the health context, applications need to conform to user privacy preserving practices [1] and the technical specifications of the data aggregation systems, e.g., the NUM-COMPASS backend. Within the scope of this work, we define conformity testing by using a set of modules in Section III. The modules aim at tackling the following challenges:

1. automated security testing of the app data flow, regarding different levels of data transmission and storage.
2. test of the FHIR resources [2] in the app at hand, regarding their conformance with the underlying data model (e.g., GECCO [3]).

We propose five modules, which break down the components of pandemic app development, and point to crucial aspects in each module focusing on automated security testing, as well as the data model-related conformance checking task. Furthermore, the problem of run-time testing is addressed, and a sketch for an *app-under-test scenario* is given in Section IV, where the focus is on the evaluation of data privacy aspects. For the conformity checking task, the usage of a certain data model consisting of FHIR resources, e.g., the GECCO data model is evaluated, for which a GECCO-approved certificate [4] was proposed within the scope of the NUM-COMPASS project. The GECCO-approved certificate aims at giving researchers, who use the NUM-COMPASS platform insight on whether the FHIR-based questionnaires and subsequently produced GECCO FHIR resources conform with the underlying data model. In Section V, we give a sketch of how the certificate may be generated within the NUM-COMPASS framework, and how versions are checked both when the app is deployed, as well as during run time.

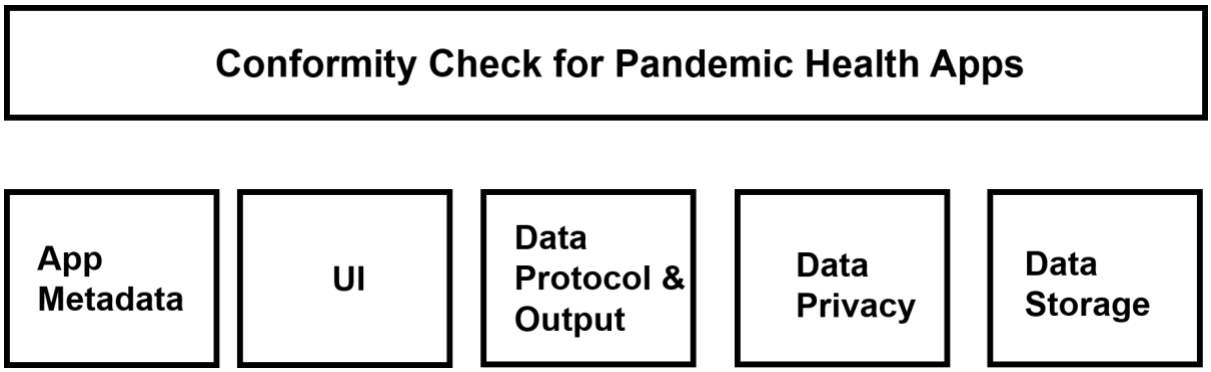

Figure 1 Block Diagram for a possible Conformity Check Test Bench

Related Work

Regarding the aspect of data traffic and automated testing of data privacy, Grundy et al. [5] proposed a method, where they selected 24 out of 821 apps identified by an app store crawling program. By using laboratory-based traffic analysis, which simulated real world app use by running four dummy scripts, the authors found that 79% of the tested apps shared user data with third party applications. The finding motivates us to point at the need of incorporating automated health app testing to the NUM-COMPASS project. Other work regarding automated testing includes leak detection [6], permission-based analysis [7], as well as man-in-the-middle attack analysis [8].

I. Modules for Conformity Checking

Figure 1 illustrates our proposed test bench to check pandemic health apps regarding their conformance with the data model, as well as privacy issues. The test bench targets different app layers using modules. Each module addresses a conformance-related task relevant to the involved app layers. Our concept reflects stages of data integration which must be tested to make sure that an app can be used in health contexts.

a. **App Metadata**
Metadata files, including the app's manifest or the Gradle file may be checked for references to the underlying data model in the App Metadata Module. The App Metadata Module is looking for traces of resources and the data model in the Gradle file or app manifest, e.g., by looking for the presence of well-known libraries developed to support expected data models, including GECCO, and protocols like FHIR.

b. **User Interface (UI)**
The UI Module of a conformance test bench deals with how the data model is incorporated into the actual user interface. Which parts of the questionnaires are used during run time? Which variables are needed and saved? How are the questionnaires displayed to the user? It may be checked, whether a smaller questionnaire, which was derived from a GECCO FHIR resource, still conforms to the GECCO FHIR data model. If a questionnaire is not in line with the GECCO FHIR resources anymore, it may be checked how it can be transformed to a data type that conforms with the GECCO data model such that this data could be sent back to a centralised database.

c. **Data Protocol and Output**
In the Data Protocol and Output Module, we look for compliance with FHIR communication standards and data structures, e.g., from the GECCO model. The layer specifically tests whether the formal requirements of the FHIR standards conform with what the app produces. To transform the questionnaire profile to a FHIR resource, the output format of the questionnaire may be determined (e.g., JSON). If all questionnaires, which are used within the app, could be transformed to profiles that conform with the FHIR data model, the test bench should provide a certificate, which tells the user that data collected from the survey can be used in a broader research context related to the data model used within the NUM-COMPASS platform, e.g., GECCO.

d. **Data Privacy**
The Data Privacy Module investigates where the variables of the data model are used and how the data is utilised during communication with other APIs and applications. The question, whether privacy-sensitive variables are sent to third-party applications without need must be answered. Private data of the user is needed if and only if it is necessary for the profiles and questionnaires defined in the data model. Pandemic-related questionnaires, for example, might demand the age or the weight of a patient, but the user might not be required to upload a profile picture. Furthermore, data privacy needs to be checked regarding the app behaviour in the network it operates in. Sensitive data shall not be sent to third-party servers or clouds if not necessary for the goal of the application.

e. **Data Storage**
In the Data Storage Module, we evaluate if stored data follows FHIR recommended practices and that storage locations follow adequate security recommendations. Two aspects are relevant here. First, it is necessary to check that only data, which is relevant for the study, are stored in the dedicated app or connected servers. Data, which is accompanied while running an application (e.g., cookies), are only to be stored when it is necessary for running the use case, for which the app is designed for. Furthermore, the servers, which are used for data storage should follow General Data Protection Regulation (GDPR) [9] standards.

II. Data Privacy: App Test During Run-Time

This section conceptualises an *app-under-test scenario* that aims at checking conformity with the data model during run time. The scenario is illustrated in Figure 2. To execute an app-under-test scenario, artificial sensor data is needed (e.g., typical acceleration sensor data, microphone data from daily activity etc.). Sensor data is fed into the app. At run time, every stage of app usage (launch, user registration, scrolling etc.) is executed, and data packages sent out to the network are checked. The test environment is split into two components that are listed in the table:

Test	Goal	Data Types
Data Tracing	How is the FHIR protocol carried out?	FHIR resources
Network Tracing	Which external clients are addressed?	Network fingerprints, e.g., IP-addresses

Within the Data Tracing test, we focus on how the FHIR protocol is carried out during run time. The test bench checks if the required resources and references are utilised in the data packages which are sent and received by the app at different stages during run time. It may be checked, if the app is directly related to the data model, e.g., GECCO, if it takes certain questionnaires and if the data was converted to a type that does not conform with the FHIR protocol anymore. If profiles of questionnaires are not in line with the underlying data model, e.g., GECCO FHIR, the test bench checks whether they could be transformed to a FHIR conformed data type. The GECCO-approved certificate is issued, if all questionnaire profiles either conform with GECCO FHIR, i.e., come directly from the GECCO data model, or could be transformed to a valid data type.

The Network Tracing test bench aims at checking how the pandemic health app communicates with different servers or clouds. To achieve a complete picture of the network environment in which the app works, a mock server is established. This mock server should communicate with the pandemic app. During different stages of app usage, IP-addresses and other fingerprints of the server should be tracked. From the addresses it shall be inferred which other servers or clouds the app sends messages or data to. The network needs to be traced with emphasis on variability, i.e., the test bench needs to verify in all different use cases and also while using the app for a long time (within the range of weeks) whether the servers it communicates with are the ones necessary for the pandemic use case at hand. This variability shall be checked with explicit emphasis on privacy issues.

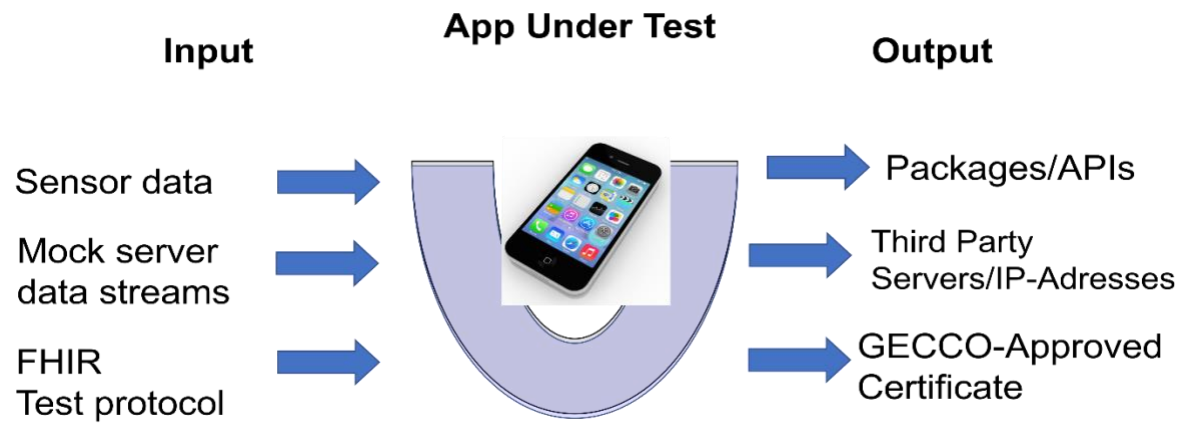

Figure 2 Schematic to illustrate how a pandemic health app could be tested during run time.

III. Version Checker for GECCO-Approved Certification

For version checking of the GECCO-approved certificate, we propose a sketch, which checks GECCO-FHIR approval in the run-time environment, as well as a NUM-COMPASS validation platform. In Figure 3, an app component, which was designed using the NUM-COMPASS framework sends GECCO questionnaires and questionnaire responses, as well as metadata and accompanying packages together with an MD5 checksum to a NUM-COMPASS validation platform. The validation platform checks the questionnaire responses for conformity with the GECCO data model, and provides packages, as well as the app version encrypted using an MD5 checksum. A signed SSL certificate is sent back to the NUM-COMPASS framework. The NUM-COMPASS framework also sends the certificate to a verification system, which checks GECCO approval during run time. Whenever the verification system approves the status of the application to be in line with the GECCO-FHIR requirements, the certificate is stored together with the app data in a dedicated database.

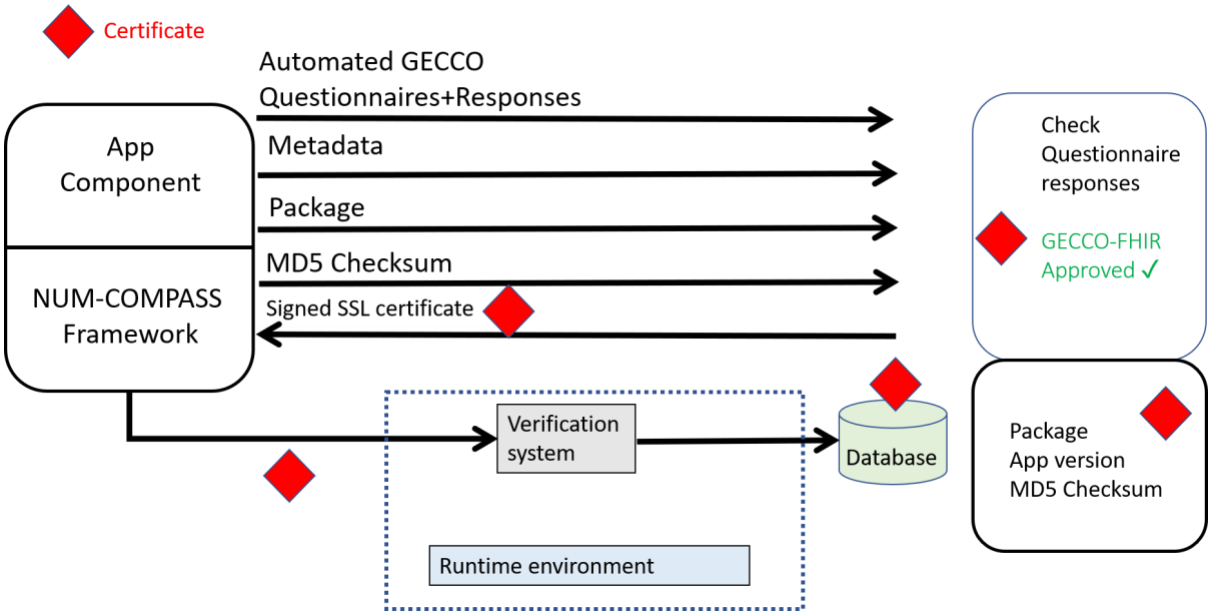

Figure 3 Sketch for a version checker approach

IV. Discussion

We proposed a set of modules, which structure conformity checking for pandemic health apps. Each module serves as a container for data model conformity checking, and an automated test bench. Data model conformity checking may result in a GECCO-approved certificate provided to the app designer. The automated test bench checks the app for unwanted data traffic with third party applications, and other privacy aspects, including GDPR regulations. Within the NUM-COMPASS project, a best practices database has been established [10], containing ideas to fill the layers with recommendations for app developers. The recommendations provided to the user may be referring to the five-layer model given in this work, and shall include both aspects of conformity checking, i.e., data model-based testing, as well as automated privacy-related tests.

V. Conclusion

This brief sketch of a possible test workflow gives a starting point for testing the GECCO data model's conformity in a pandemic app context. To proceed with the work, available protocols for conformity checking may be investigated. Furthermore, the GECCO data model's output variables, and its integration to the app manifest shall be further investigated. To assure that privacy is not lost for the user, unwanted communication with third party applications should be prevented.

Referenzen

[1] Psychoula, I., Chen, L., and Amft, O. *Privacy Risk Awareness in Wearables and the Internet of Things*. IEEE Pervasive Computing 19, 60–66, 2020.

[2] M. L. Braunstein, *Health Informatics on FHIR: How HL7's New API Is Transforming Healthcare*. Cham: Springer International Publishing, 2018.

[3] J. Sass, A. Bartschke, M. Lehne, A. Essenwanger, E. Rinaldi, S. Rudolph, K. U. Heitmann, J. J. Vehreschild, C. von Kalle, and S. Thun, "The German Corona Consensus Dataset (GECCO): A standardized dataset for COVID-19 research in university medicine and beyond," *BMC Medical Informatics and Decision Making*, vol. 20, p. 341, Dec. 2020.

[4] M. R. Muzoora, N. El-Badawi, C. Elsner, A. Essenwanger, P. Gocke, D. Krefting, R. A. Poyraz, R. Pryss, U. Sax, and S. Thun, "Motivating Developers to Use Interoperable Standards for Data in Pandemic Health Apps," *Studies in Health Technology and Informatics*, vol. 281, pp. 1027– 1028, May 2021.

[5] Q. Grundy, K. Chiu, F. Held, A. Continella, L. Bero, and R. Holz, "Data sharing practices of medicines related apps and the mobile ecosystem: Traffic, content, and network analysis," *BMJ*, vol. 364, p. l920, Mar. 2019.

[6] A. Continella, Y. Fratantonio, M. Lindorfer, A. Puccetti, A. Zand, C. Kruegel, and G. Vigna, "Obfuscation-resilient privacy leak detection for mobile apps through differential analysis," *Proceedings of the ISOC Network and Distributed System Security Symposium (NDSS)*, Feb. 2017.

[7] Tiwari and U. Singh, *Android Users Security via Permission Based Analysis*. Aug. 2015.

[8] K. Huckvale, J. T. Prieto, M. Tilney, P.-J. Benghozi, and J. Car, "Un- addressed privacy risks in accredited health and wellness apps: A cross- sectional systematic assessment," *BMC Medicine*, vol. 13, Sept. 2015.

[9] "General Data Protection Regulation (GDPR) Compliance Guidelines." https://gdpr.eu/.

[10] "NUM Compass." https://num.umg.eu/.

Folgende Universitätskliniken des Netzwerks Universitätsmedizin nehmen am COMPASS-Projekt teil:

Charité – Universitätsmedizin Berlin
Universitätsmedizin Göttingen
Universitätsmedizin Mainz
Universitätsklinikum Würzburg
Uniklinik Köln
Universitätsklinikum Münster
Universitätsklinikum Regensburg
Universitätsklinikum Ulm
Universitätsklinikum Erlangen

Ansprechpartner für weitere Fragen:
COMPASS Koordinierungsstelle
compass@unimedizin-mainz.de

num compass

https://num-compass.science

@CompassNum

netzwerk universitätsmedizin

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung