

# StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

Hannah A. Burkhardt<sup>1\*</sup>, Pascal S. Brandt<sup>1</sup>, Jenney R. Lee<sup>1</sup>, Sierramatice W. Karras, Paul F. Bugni, Ivan Cvitkovic, Amy Y. Chen, William B. Lober

<sup>1</sup>University of Washington, Seattle, WA

## Abstract

As the COVID-19 pandemic continues to unfold and states experience the impacts of reopened economies, it is critical to efficiently manage new outbreaks through widespread testing and monitoring of both new and possible cases. Existing labor-intensive public health workflows may benefit from information collection directly from individuals through patient-reported outcomes (PROs) systems. Our objective was to develop a reusable, mobile-friendly application for collecting PROs and experiences to support COVID-19 symptom self-monitoring and data sharing with appropriate public health agencies, using Fast Healthcare Interoperability Resources (FHIR) for interoperability. We conducted a needs assessment and designed and developed StayHome, a mobile PRO administration tool. FHIR serves as the primary data model and driver of business logic. Keycloak, AWS, Docker, and other technologies were used for deployment. Several FHIR modules were used to create a novel “FHIR-native” application design. By leveraging FHIR to shape not only the interface strategy but also the information architecture of the application, StayHome enables the consistent standards-based representation of data and reduces the barrier to integration with public health information systems. FHIR supported rapid application development by providing a domain-appropriate data model and tooling. FHIR modules and implementation guides were referenced in design and implementation. However, there are gaps in the FHIR specification which must be recognized and addressed appropriately. StayHome is live and accessible to the public at <https://stayhome.app>. The code and resources required to build and deploy the application are available from <https://github.com/uwcirg/stayhome-project>.

**Keywords:** COVID-19, Health Information Interoperability, Patient-reported outcomes, Mobile Applications, Epidemiological monitoring

Correspondence: [haalbu@uw.edu](mailto:haalbu@uw.edu)\*

DOI: 10.5210/ojphi.v13i1.11462

Copyright ©2021 the author(s)

This is an Open Access article. Authors own copyright of their articles appearing in the Online Journal of Public Health Informatics. Readers may copy articles without permission of the copyright owner(s), as long as the author and OJPHI are acknowledged in the copy and the copy is used for educational, not-for-profit purposes.

## Introduction

SARS-CoV-2, a novel coronavirus which causes the disease COVID-19, was first reported in December 2019 in China [1] and quickly spread globally, causing millions to contract the virus and fall ill. In the US, the disease was first discovered in a Washington State nursing home in February 2020. As of August 2020, the virus has claimed over 170 thousand lives in the US [2]. Starting in April 2020, states imposed severe limitations on businesses and individuals in an attempt to curb spread. As of August 2020, states have relaxed stay-at-home orders and are experiencing diverse outcomes from reopened economies. Blueprints to minimize the increased outbreaks include public health efforts such as widespread testing, identification of cases, and intensive contact tracing [3]. States are rapidly scaling up the number of individuals conducting contact tracing investigations to meet this need, which means that the contact tracing workforce is comparatively inexperienced, yet meets with high caseloads. An additional exacerbating factor is that many sources of essential data, such as community-based testing, medical records, and public health records are managed in ways that inhibit their effective linkage. Informatics solutions may ease and support this labor-intensive, manual work. We can draw on the experiences of patient-centered systems in clinical informatics, which recognize that often the patient and their reported experiences may be the best links between information managed in disparate locations and systems. In the context of COVID-19, this may include events such as testing and clinical visits, as well as personal information such as contacts and travel, and clinically relevant symptom monitoring.

Individuals are encouraged to wash their hands, wear masks, maintain physical distance from others, and self-monitor for symptoms in order to recognize when further steps become necessary. Proactively tracking symptoms may allow recognition of emerging symptomatology, allowing earlier self-isolation and reducing transmissions, and is therefore a core need of the community. Maintaining a diary of symptoms, possible exposures, virus and antibody testing, and travel history has the additional potential benefit of providing public health agencies with reliable information to conduct case investigations, should that need arise.

The more people keep and are willing to share such records, the greater the benefit may be to the community. Symptom trackers and exposure diaries should therefore be easy to use and accessible to everyone, including vulnerable populations. Choosing a mobile-accessible design supports this goal due to the ubiquity of mobile phones and the potential to reach more individuals from vulnerable communities. In 2019, over 80% of individuals and over 50% of older adults in the United States owned a smartphone; 17% had internet access only on their phone, with higher proportions for poor populations (26%) as well as Black (23%) and Hispanic (25%) minority groups [4].

Individuals should be able to share diaries of COVID-relevant observations directly and easily with the relevant agencies if they so choose. Using health data standards can lower the barrier to sharing symptom tracker and exposure diary data with third parties. The Fast Healthcare Interoperability Resources (FHIR) [5] standard may facilitate data sharing, both with electronic health records (EHRs) for use in clinical care, and with public health to benefit contact tracing.

## Representative COVID-19 apps

The COVID-19 pandemic has prompted the development of numerous consumer applications intended to support individuals, to complement traditional public health surveillance efforts, or both. Some examples of these apps, grouped into broad categories, are shown in Table 1. As a longitudinal symptom tracker, StayHome represents a less common application type, in contrast with “low-tech” symptom screeners and “high-tech” contact tracing apps. Yet, there have been examples of such applications seeing high adoption and even enabling disease research: A symptom tracker launched in the UK on March 24, 2020 (“COVID Symptom Study App”) gained 700,000 users within 24 hours and ultimately contributed to the reporting that loss of sense of taste or smell is associated with COVID-19 infection [6-8].

StayHome first and foremost supports self-monitoring by individuals. Supporting public health efforts is a secondary goal enabled by StayHome if a need arises.

**Table 1. Comparison of COVID app modalities**

	<b>Description</b>	<b>Example</b>	<b>Data collected</b>	<b>Longitudinal</b>
<b>Symptom checker/ screener</b>	Allows the user to enter their symptoms and receive medical advice.  Uses the responses to recommend one of several possible next steps, e.g. continuing to practice social distancing or contacting a healthcare provider.	Apple’s “COVID-19 Screening Tool” [9]  CDC’s “Symptom Self-Checker” [10]	Symptoms associated with COVID-19 (e.g. fever, cough), risk factors (e.g. age, comorbidities), recent travel, possible exposures.	No. Applications are designed for one-off use and do not collect data longitudinally.
<b>Contact tracing apps</b>	Keeps track of people users have been in contact with by collecting location and proximity data, and alerts users of possible exposures. May utilize a range of	BlueTrace (Singapore) [11]  Corona-Warn-App (Germany) [12]	Collects records of which other devices a device was near over the past days or weeks, enabling alerting of possibly exposed individuals in	Yes.  Applications track location or proximity information over time.

	technologies, e.g. Bluetooth, GPS, or Google and Apple’s joint API.		the case of a positive test.	
<b>Longitudinal symptom tracker/contact diary</b>	Helps users track symptoms, possible exposures, and other data over time, potentially enabling discovery of longitudinal trends.	COVID Symptom Study App (UK) [7] StayHome	Symptoms associated with COVID-19 (e.g. fever, cough), risk factors (e.g. age, comorbidities), recent travel, possible exposures.  Data is collected repeatedly (e.g. once daily)	Yes. Applications track symptoms and other data over time.

**FHIR**

Lack of standardization can both delay access and reduce the quality of the data available to public health agencies. To address broad issues of health data interoperability, Health Level Seven International (HL7) has published the FHIR standard. Applicable healthcare organizations are required to implement and maintain the FHIR application programming interface (API) per the Interoperability and Information Blocking Rule of the 21st Century Cures Act (effective June 2020) [13,14]. FHIR describes a standard representation for many common entities in the healthcare domain, defines relations between these entities, and describes a variety of computational methods for operating on these entities. The FHIR specification describes recurrent healthcare application business requirements in dedicated modules, accessible via the documentation index on the FHIR website (<https://www.hl7.org/fhir/>). Two examples are the Workflow Module and the Clinical Reasoning Module. The Workflow Module describes how care plans and specific care-related workflows can be characterized, scheduled, and executed. The Clinical Reasoning Module provides a mechanism to represent and evaluate clinical knowledge in an entirely FHIR-native way, that is, using FHIR-compliant data structures, operations, and logical expression languages like FHIRPath [15] and the Clinical Quality Language (CQL).

The FHIR specification is flexible in how it can model data and workflows. Supplementary standard operating procedures and constraints are published in Implementation Guides (IGs), supplying additional details and restrictions required for true interoperability, and offering guidance on how to use FHIR to solve particular problems. For example, the US Core IG places restrictions on entity attributes and terminologies (e.g. ICD-10-CM). Additionally, the Structured Data Capture (SDC) IG guides the interoperable and FHIR-compliant implementation of data entry forms. Given the broad nature of the FHIR specification and the existence of mature IGs that

address the requirements of the StayHome application, there is an opportunity to use FHIR as the underlying model for both application data and business logic.

## Significance

To address the need for community-based self-monitoring and exposure tracking, to support individuals' decision making and contact tracing efforts in case of infection, we developed StayHome, a reusable, mobile-friendly, longitudinal symptom tracker designed and developed following user-centered design principles. StayHome allows regular logging of symptoms and other information, and review of data over time.

StayHome is unique in that it is implemented not just to exchange data using FHIR, but to adopt FHIR resources for internal representation of data and business logic, a design approach we call "FHIR-native". Traditionally, FHIR and its predecessors, such as the HL7v2 messaging standard, are used first and foremost to support an interoperable interface strategy. In line with this primary goal of FHIR, StayHome's use of the standard enables interoperability with other health informatics systems, such as electronic health record (EHR) systems and public health informatics systems. However, StayHome also leverages FHIR as internal information architecture, using the standard's domain models to represent both data and business logic. This holistic use of FHIR makes the application a generic PRO tool independent from any specific health problem, PRO use case, or host system.

StayHome is open source and freely available to anyone to use, modify, and implement for clinical and consumer health informatics applications (under the BSD 3-Clause license) at <https://github.com/uwcirg/stayhome-client> and associated repositories.

With FHIR adoption increasing, informaticists are exploring how best to use it to address important issues in health informatics. We share lessons learned from developing and deploying this application, which we hope will benefit others in designing and developing applications using this new standard.

## Methods

### Needs Assessment & Design

In February 2020, before Washington State's stay-home-orders were put in place and before the outbreak was officially characterized as a pandemic, we recognized the potential benefits of a symptom tracker application. To inform the requirements for such an application, we solicited input from students in an undergraduate class about science, evidence, and health, conducted by one of the authors (WL), where COVID-19 had become a frequent topic of discussion. This process included informal conversations with the students as well as a structured survey asking participants about their COVID-19 concerns, and how a smartphone application might help them with those concerns (HB, WL, SK). Project team members (SK, HB, WL) then tabulated survey results and used content analysis methodology to identify and prioritize user-centered design features. The University of Washington IRB determined on July 6, 2020 that Ethics approval was not required for this project.

## StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

With the pandemic actively unfolding, there was an immediate need for the application. We therefore aimed to keep the user experience simple, using standard mobile UI components and interaction patterns where possible, while maintaining a user-centric approach to application design by conducting user tests and revising based on feedback in an iterative fashion.

### **FHIR**

In addition to using FHIR resources to represent our data model, we also made use of the FHIR RESTful API [16] to create and update both metadata (e.g. Questionnaire resources) and data (e.g. Patient resources) and the FHIR Search API for data retrieval. Concepts from the Workflow Module (e.g. definitional resources) were used for PRO workflow execution. We dynamically encoded logic via FHIRPath expressions as part of questionnaire display items, per the Clinical Reasoning Module and SDC guidance. Internationalization was implemented using guidance from the Implementation Support Module. The Consent resource was employed in conjunction with guidance from the Security and Privacy Module to record data sharing preferences. We referenced the Terminology Module to implement a custom code system for app-internal messages/notifications. Extended operations, as described in the Exchange Module, were used to expand answer option value sets.

### **Development and Deployment**

Mobile applications exist within several disjoint consumer app ecosystems characterized by different hardware, software, and process constraints, which is a barrier to mobile app development and maintenance. Cross-platform mobile application development frameworks are available to address this challenge, allowing the use of a single codebase for developing Android, iOS, and web applications. This work utilized Google's Flutter [17] cross-platform mobile application development framework.

Figure 1 shows an overview of the overall system architecture. In addition to the client applications (community facing and administrative), the system includes Keycloak [18] as an Open ID Connect identity provider, a reverse proxy for role-based access control ("map-api"), and HAPI FHIR version 4.2.0 as a FHIR API server and persistent data store.

Amazon Web Services (AWS) cloud infrastructure provides high availability and scalability for each server component, and Docker supports continuous integration (CI) and deployment. StayHome is internationalized to English, Spanish, Haitian Creole, and German, with automated string export and import processes.

GitLab and GitHub were used for version control and to make our work publicly accessible. The full code for the StayHome app, the authorization reverse proxy server, and Docker files needed for deployment are open-source and freely available.

# StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

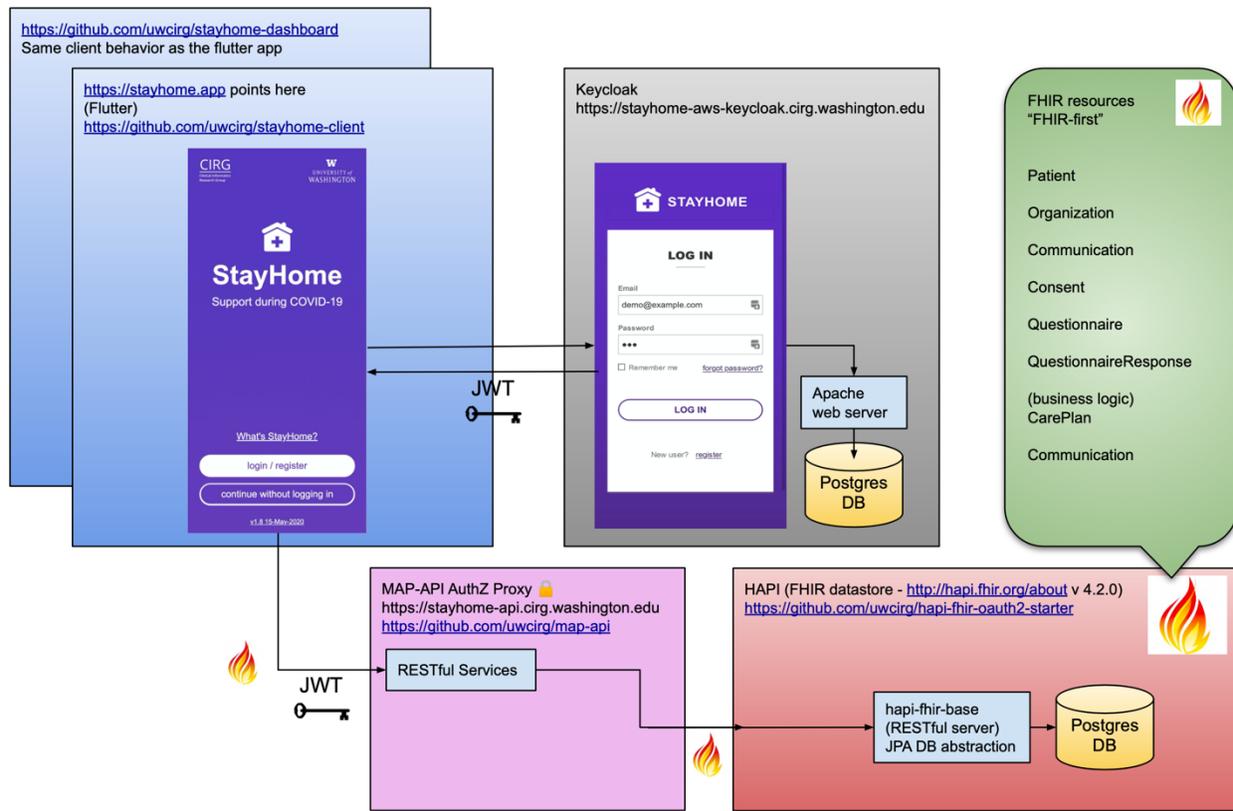


Figure 1. System architecture. In addition to the client applications (community facing and administrative), the system includes Keycloak as an identity provider, a reverse proxy for role-based access control (“map-api”), and HAPI FHIR version 4.2.0 as a FHIR API server and persistent data store.

We began software development in February 2020 and published version 1.0 on March 27, 2020. StayHome is live and accessible to the public at <https://stayhome.app>.

A companion dashboard application (“stayhome-dashboard”) was developed to allow administrators to view and manage data and users. This application supports public health surveillance and reporting workflows by providing search and filter functionality and allowing review of usage statistics and user level-information such as location (for users who choose to share that information). The appearance of user data from the client-app is based on “opt-in” permissions given by the user of the client app. Further, the dashboard provides a venue to host future functionality, such as advanced reporting and data visualization.

## Results

### Needs Assessment & Design

There were 63 undergraduate students enrolled in the class; about 40 students were present in class when we conducted the assessment in February 2020. Participants indicated that they were concerned about infection and developing the disease and that they were unsure of how to protect

# StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

themselves and others. Many were wondering how they would be able to tell if they had contracted the virus. Some participants indicated that they measured their body temperature daily. With word of mouth being a major information source and official sources recommending caution or even de-emphasizing the seriousness of the outbreak, participants expressed uncertainty that fueled a need for reliable information. Based on these conversations with potential users, it became apparent that we had the opportunity to develop an application to address the need for daily self-monitoring of symptoms and clinical signs.

We aimed for a simple, functional design. However, navigation design was challenging and the subject of several design iterations. Our goal was to streamline the user experience so users could quickly find frequent actions, while still allowing easy access to less-frequently used functions. For example, the symptom assessment questionnaire might be completed daily. In contrast, the risk factors and exposures questionnaires would only be filled out once (or in the case of a relevant event). Additionally, we wanted to avoid confusion and provide reassurance in how data would be collected and used, so we opted to display lay-person-friendly terms and conditions and detailed explanations in-line with the controls that asked for such information. Figure 2 shows screenshots of the current user interface.

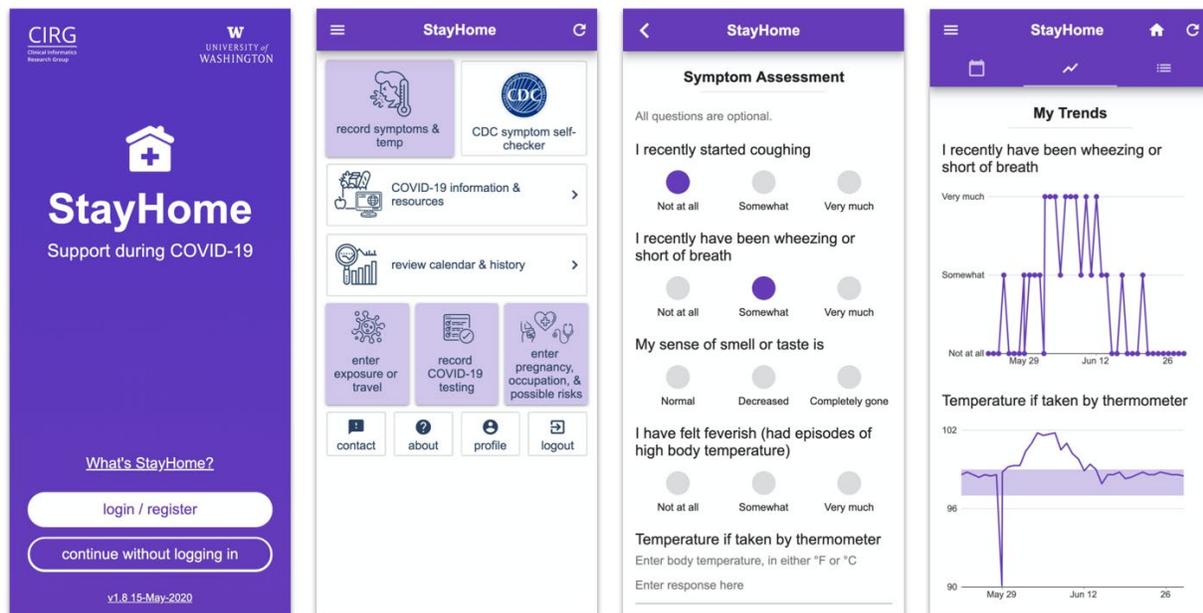


Figure 2. StayHome application UI. From left to right: Login/start screen, home screen, questionnaire screen, trends review screen.

## FHIR

The use of FHIR supported the rapid development of the StayHome application by providing a domain-appropriate data model and query framework. FHIR specifies the general structure (i.e. applicable data elements) of resources describing various concepts in health and healthcare applications; there is, however, significant flexibility in how these resources can support individual use cases. This section gives an overview of some of the parts of the FHIR specification used, and

how they support the StayHome use case. See Figure 3 for an overview of how resources work together.

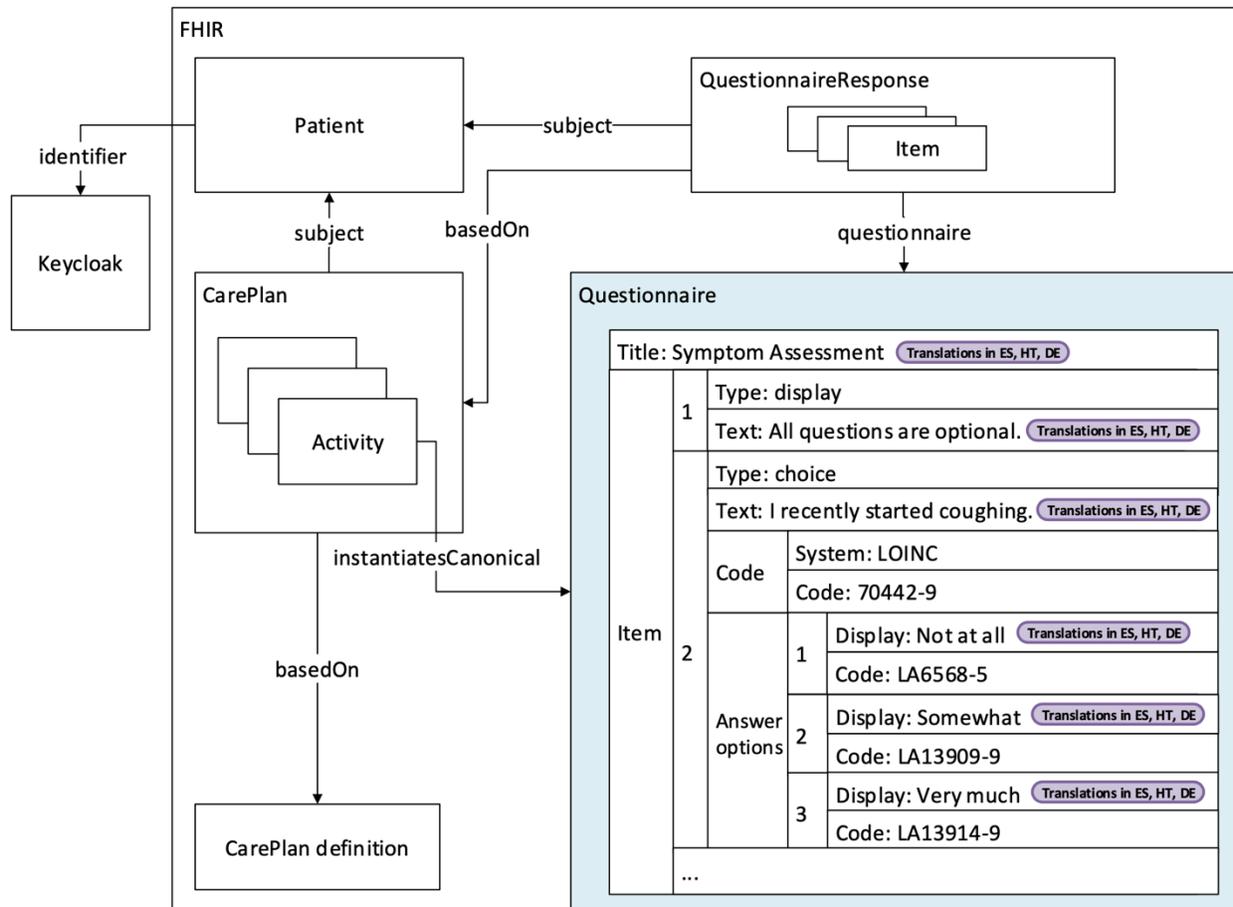


Figure 3. FHIR resources used by StayHome.

### Resources

**Patient.** The patient resource links individual users' personal information and settings with their account login managed by the external identity provider service. StayHome uses **Patient** resources to manage demographics, contact information, and language preference (the application does not require users to enter this information).

**CarePlan.** This resource specifies a set of questionnaires. A definitional resource is defined on the application level, which is then instantiated for each new user. Instantiation allows subsequent, person-specific adjustment of the set of questionnaires to be administered and provides the opportunity to personalize the frequency of individual instruments. CarePlan resources reference the patient they belong to via the subject element and list the questionnaires in the activity element.

**Questionnaire.** This resource defines the content of each PRO instrument and is thus central to StayHome. It consists of a list of display items, which can be either an answerable question or help

## StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

text. Each item includes answer options coded for interoperability (e.g. LOINC codes), selection behavior (e.g. single answer vs. multiple answer), and formatting information (e.g. section headers vs. help text), as appropriate. Items can also represent calculated quantities (e.g. score totals), conditionally displayed items to support branching logic, and conditional user messages (e.g. a “high risk” and a “low risk” message, presented based on the calculated score total). For an excerpt from the symptom assessment questionnaire, see Figure 3. QuestionnaireResponse resources record responses to questionnaires.

**Consent.** StayHome enables users to indicate their data sharing preferences using Consent resources. Each of these resources indicates whether an individual user consented to share their data with the stated organization.

**Organization.** This resource type is used to identify entities to whom sharing consent can be given, e.g. public health agencies, researchers, and potential community partners, and is referenced by Consent resources.

### *Query API*

StayHome utilizes FHIR’s built-in search APIs in several ways. First, resources are queried by ID in cases where that ID is known, e.g. in CarePlans, which specify their component Questionnaires by ID. Second, the application retrieves relevant resources with simple search queries, e.g. to retrieve the Patient resource corresponding to the logged-in user. Finally, multiple combined search and filter criteria are used to retrieve bundles of resources for specific use cases; for example, to plot response history for individual questions. The administrative dashboard uses such a query to retrieve records. The “map-api” reverse proxy ensures that only resources with proper authorization access are returned, i.e. those belonging to the logged-in user, or on the dashboard, those referring to patients who consented to share data.

### *Internationalization*

To display questionnaires in different languages according to user choice, we utilized both the Translatable and Translation extensions following guidance from the Implementation Support Module (Internationalization section). The Translatable extension tags strings as user-facing. We added an automatic extraction step to push untranslated English strings to the CI pipeline. After receiving translations from translators working in translation software such as Transifex [19], translations were inserted back into the FHIR resources using the Translation extension (Code snippet 1).

**Code snippet 1. JSON example of a string element with two translations. The string is tagged as user-facing with the Translatable extension.**

```
"text": "Start date",
  "_text": {
    "extension": [
      {
        "url": "http://hl7.org/fhir/StructureDefinition/elementdefinition-translatable",
```

```
"valueBoolean": true
},
{
  "url": "http://hl7.org/fhir/StructureDefinition/translation",
  "extension": [
    {
      "url": "lang",
      "valueCode": "de"
    },
    {
      "url": "content",
      "valueString": "Anfangsdatum"
    }
  ]
},
{
  "url": "http://hl7.org/fhir/StructureDefinition/translation",
  "extension": [
    {
      "url": "lang",
      "valueCode": "es"
    },
    {
      "url": "content",
      "valueString": "Fecha de inicio"
    }
  ]
}
],
},
]
```

### *Calculated Questionnaire Items*

The FHIR Structured Data Capture (SDC) IG describes how to use expressions and extensions to support calculated display items, such as score totals. StayHome employs FHIRPath as an expression language, and the ordinalValue extension attaches numeric values to categorical response options. The freely available fhirpath.js [20] library is used to calculate the value of the expressions in real-time as the user enters questionnaire responses. The Questionnaire resource includes the expression shown in Code snippet 2 for this purpose.

### **Code snippet 2. FHIRPath expression used by StayHome to calculate the sum of all selected answers' ordinal values (score total).**

```
QuestionnaireResponse.item.answer.extension.where(url.contains('valueOrdinal')
).valueDecimal.aggregate($this + $total, 0)
```

## **Discussion**

In this work, we designed, developed, and deployed a mobile-friendly, FHIR-native, PRO tool with application to COVID-19 symptom assessment and exposure diary functions. This

information system administers questionnaires and organizes information using FHIR for the representation of both application content and behavior.

## **FHIR-Native Approach**

### ***Advantages***

Using FHIR in general and in a FHIR-native application context has significant advantages (Table 2). The FHIR standard emerged from the modeling work of many experts and thus provides an excellent starting point for an application-specific data model in the health domain. While our application's needs are comparatively simple, starting with data structures capable of accommodating advanced data and process models obviated the need to restructure the data model in the face of new business requirements. Existing FHIR server implementations such as HAPI provide routine functionality, such as create, read, update, and delete (CRUD) APIs, as well as advanced domain-specific tooling, such as CQL support, allowing developers to focus on their application's unique requirements. Because FHIR specifies the interface between clients and servers, implementations are interchangeable as long as they comply with this interface. This allows the operation of apps directly against either standalone FHIR servers or other FHIR-compliant servers, e.g. those belonging to EHRs or public health information systems. FHIR also facilitates the exchange of information between systems, known as interoperability, though carefully following implementation guides is required to achieve semantic interoperability.

Beyond data exchange, we leveraged FHIR as the internal information architecture. This approach facilitated rapid application development by reducing the need to design and refine a data model, as well as the tooling to support it. Because FHIR resources (e.g. definitional resources and questionnaires) drive application content and behavior, PRO functionality is reusable for any FHIR-compliant questionnaire. The application requires no custom server code and operates entirely within the FHIR specification, facilitating the reuse of StayHome as a generic FHIR-enabled questionnaire administration tool. FHIR can accommodate complex business logic in an elegant, standardized way, making possible a wide range of applications that require no server code.

### ***Disadvantages***

There are disadvantages to using FHIR, which may intensify for FHIR-native applications. Though the ability to use existing server implementations and tools offsets the time investment required to use FHIR, the complexity and size of the FHIR specification and its extended documents may be a barrier to adoption. Additionally, in some domains, e.g. in the consumer health application realm, FHIR may not be widely adopted, reducing the immediate benefit of interoperability-readiness. Further, many use cases, clinical and non-clinical, are not modeled by FHIR. While the standard is flexible and extensible to new scenarios, there is a risk of tipping the balance from using it appropriately to counterproductively shoehorning application data and business logic into FHIR. For example, if resources or resource elements cannot be understood outside of the StayHome context or are not compatible with recommendations laid out in commonly used IGs such as US Core, the use of FHIR may turn out to be a burden.

Finally, despite the guidance from FHIR modules and IGs, many implementation and representation choices remain with the developer, and these design choices may be imperfect. For example, we considered using a single, unmodifiable CarePlan resource to define the set of questionnaires at the application level. We instead chose to instantiate a user-level CarePlan resource with reference to an application-level definitional resource (template) to allow personalization of app content. This instantiation introduced additional barriers for content maintenance: adding new questionnaires to the definitional resource required updating every user’s CarePlan instance as well (previous versions are still accessible via the history API). This experience underscored the need to make representational choices carefully and with specific use cases in mind; to ground decisions in implementation guides, which address many common difficulties, as much as possible; and to accept the fact that updating large numbers of resources as part of application updates may ultimately be unavoidable as business requirements evolve.

**Table 2. FHIR/FHIR-native advantages and disadvantages.**

Pros	Cons
<ul style="list-style-type: none"> <li>● Represents modeling work of many experts</li> <li>● Specifies common requirements and corresponding tools/APIs (e.g. storage functionality, queries, versioning, CRUD operations, data formats, localization)</li> <li>● Specifies advanced functionality; specific implementations (e.g. HAPI) might provide further functions that become automatically available to FHIR-native apps (e.g. Async to export bulk data, terminology services to translate codes, advanced searches, subscriptions, CQL)</li> <li>● In theory, EHR workflow integration out of the box with SMART [21]</li> <li>● In principle, standard data format enables immediate interoperability with other systems: Resources can be consumed by other apps and other apps’ resources could be consumed by StayHome</li> <li>● IGs help with solving common challenges and further semantic interoperability</li> <li>● FHIR implementations are interchangeable (e.g. HAPI, Azure and Google Cloud FHIR servers, EHR with FHIR APIs, public health system with FHIR APIs)</li> </ul>	<ul style="list-style-type: none"> <li>● Complex: it can be challenging to determine the intent of different pieces, and what applies to a particular use case; takes time to work through; developing compliant applications requires consideration of constraints from different places</li> <li>● Specific application constraints and requirements may not have been considered when developing the spec</li> <li>● Does not specify all functionality required for a complete app, e.g. access control, workflows for non-clinical use cases</li> <li>● FHIR does not model some specialized/complex clinical use cases</li> <li>● Risk of forcing data and business logic into FHIR in a way it wasn’t intended for, compromising interoperability and application reusability in practice</li> <li>● May currently be limited in some application domains, e.g. wearables and other consumer health applications</li> </ul>

<ul style="list-style-type: none"><li>● Clients are independent of the server / can operate out of any FHIR-compliant server (e.g. a standalone system or an EHR).</li><li>● PRO functionality is reusable to any questionnaire that can be specified as FHIR</li><li>● Built-in backward compatibility (between the data model and different versions of client apps)</li><li>● Standard terminologies enable translating the application to work in a new context (e.g. to use LOINC instead of SNOMED)</li></ul>	
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

## Development and Deployment

For our user-facing application, we chose Flutter as a solution to cross-platform mobile application development, but quickly felt the impact of using emerging technologies. For example, there was no FHIR resource library for the Dart programming language, requiring us to generate and test the data model and serialization code. On the other hand, using an actively evolving framework enabled us to take advantage of cutting-edge technology. For example, Flutter Web was first released in the beta channel in December 2019. Not long before our initial release, we recognized that our group did not have the necessary resources and expertise to overcome the challenges of mobile app publishing (e.g. registering as a non-profit entity, integrating iOS application builds in our CI pipeline), within an acceptable time frame. Flutter Web, while suffering idiosyncrasies typical of beta-stage software, was an alternative that became available just in time for us to publish StayHome as a web application instead.

Web applications have many benefits over native applications. They provide the same experience on different platforms, including desktop, maximizing reach while minimizing the additional development effort required to support many different platforms. Deployments (such as when a new version is published) affect end-users minimally, as client browser apps will automatically retrieve the latest version of the web application when reloading, eliminating the need to update local application installs. Additionally, some development teams may find deploying a web application to be straightforward compared to completing the steps required to get a native mobile application approved and published on each of the major platform app stores. While web applications may provide suboptimal UX compared to native apps, there is an opportunity to narrow that gap via Progressive Web Application (PWA) or “installable web application” setups. Other drawbacks of non-native applications include limitations for integration with local hardware (e.g. Bluetooth, camera) and barriers in using notifications.

Authentication and authorization for FHIR were challenges, as fine-grained data access control is an infrastructure requirement not adequately addressed by FHIR. While solutions such as SMART on FHIR and Interceptors for HAPI FHIR were a possibility for authentication and authorization, we instead implemented a lightweight reverse proxy server as a single solution for both authentication and authorization. This approach removed the need to manage and secure

## StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

credentials within the StayHome app and allowed us to keep the server implementation (e.g. HAPI vs. EHR) interchangeable and independent of the client app. While comparatively simple conceptually and in practice, this “non-standard SMART on FHIR” setup adds a custom layer in our system architecture. We are actively investigating the use of a so-called standalone SMART on FHIR application setup to simplify our authorization system.

### **Future work**

Future work includes implementing a fully SMART compliant launch flow. Additionally, there is an opportunity to further leverage FHIR to support more advanced functionality, for example, by using extraction as described in the SDC module to create Observation resources from QuestionnaireResponse resources.

While we provided the application for use to members of our immediate community to support community members as soon as possible and to collect further feedback, accomplishing broad adoption of the app requires resources and expertise, including marketing and consumer support, that lie outside of our capabilities and resources. However, the application may be useful for specific programs, such as research projects and community health programs, with well-defined objectives and user groups. Such small- to medium-scale implementations thus represent an important future direction for this work.

### **Conclusion**

We presented StayHome, a FHIR-native PRO tool applied as a COVID-19 symptom tracking application and public health reporting tool. We deployed this tool in a software system that includes a community member-facing web application, authentication/authorization layer, FHIR server, and administrative dashboard application. Code and resources are open source under the terms of the BSD 3-Clause license. As a mobile-friendly web application, StayHome maximizes reach. StayHome addresses core community needs by providing users with a tool to self-track possible symptoms and exposures and supports them in following public health recommendations for their personal health. The use of FHIR enables user-controlled interoperability with, for example, public health contact tracing and case investigation systems, an area the authors are actively working on with the Washington State Department of Health through the CommonCircle initiative. We innovatively operationalized FHIR beyond its use as a messaging standard, leveraging it as an internal information architecture to represent application content and behavior. Thus, we created a questionnaire administration tool that is easily reusable for other use cases by simply updating FHIR resources. The tool can operate out of any host system, e.g. a standalone FHIR server, an EHR, or another FHIR-compliant server. During this process, we found that there are advantages and disadvantages in the use of FHIR and the FHIR-native approach. We applied FHIR in a novel way by using it as internal information architecture, but also found that the right level of FHIR use can be a balancing act that depends on individual application requirements and constraints. We used emerging technologies, tried-and-true technologies, and custom system components, finding benefits and tradeoffs for each. We also recognized that pre-existing expertise is a factor in deployment considerations. We provide our code in several open-source repositories, freely available under the BSD 3-Clause license at <https://github.com/uwcirg/stayhome-client> and in associated repositories. We hope that the StayHome application benefits the community in

working to stay safe and flatten the curve. Additionally, we hope that the code and resources found in our Github repositories and the considerations shared in this manuscript benefit others in their efforts to develop and operationalize FHIR applications in general and PRO applications in particular.

## Acknowledgements

The authors conducted this work through the UW Clinical Informatics Research Group (CIRG), and gratefully acknowledge the contributions of CIRG staff, especially Justin McReynolds.

## Funding

This work was partially supported by the National Library of Medicine Training Grant (T15LM007442).

## References

1. Timeline of WHO's response to COVID-19. Accessed July 1, 2020. <https://www.who.int/news-room/detail/29-06-2020-covidtimeline>
2. Dong E, Du H, Gardner L. 2020. An interactive web-based dashboard to track COVID-19 in real time. *Lancet Infect Dis.* 20(5), 533-34. doi:[https://doi.org/10.1016/S1473-3099\(20\)30120-1](https://doi.org/10.1016/S1473-3099(20)30120-1). [PubMed](#)
3. McClellan M, Gottlieb S, Mostashari F, Rivers C, Silvis L. A national COVID-19 surveillance system: Achieving containment. Published online 2020.
4. Andersen M. Mobile Technology and Home Broadband 2019 | Pew Research Center. Published online 2019. Accessed June 9, 2020. <https://www.pewresearch.org/internet/2019/06/13/mobile-technology-and-home-broadband-2019/>
5. Bender D, Sartipi K. HL7 FHIR: An agile and RESTful approach to healthcare information exchange. In: *Proceedings of CBMS 2013 - 26th IEEE International Symposium on Computer-Based Medical Systems.*; 2013:326-331. doi:10.1109/CBMS.2013.6627810
6. Mayor S. 2020. Covid-19: Researchers launch app to track spread of symptoms in the UK. *BMJ.* 368, m1263. doi:<https://doi.org/10.1136/bmj.m1263>. [PubMed](#)
7. Drew DA, Nguyen LH, Steves CJ, et al. Rapid implementation of mobile technology for real-time epidemiology of COVID-19. *Science (80-)*. Published online May 5, 2020:eabc0473. doi:10.1126/science.abc0473
8. Menni C, Valdes AM, Freidin MB, et al. Real-time tracking of self-reported symptoms to predict potential COVID-19. *Nat Med.* Published online May 2020. doi:10.1038/s41591-020-0916-2. [PubMed](#)

## StayHome: A FHIR-Native Mobile COVID-19 Symptom Tracker and Public Health Reporting Tool

9. Coronavirus (COVID-19) - Apple and CDC. Accessed August 17, 2020. <https://www.apple.com/covid19/>
10. Symptoms of Coronavirus | CDC. Accessed August 17, 2020. <https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html>
11. Bay J, Kek J, Tan A, et al. BlueTrace : A privacy-preserving protocol for community-driven contact tracing across borders. *Gov Technol Agency, Singapore*. Published online 2020:9. [https://bluetrace.io/static/bluetrace\\_whitepaper-938063656596c104632def383eb33b3c.pdf](https://bluetrace.io/static/bluetrace_whitepaper-938063656596c104632def383eb33b3c.pdf)
12. Corona Warn App from SAP & Deutsche Telekom | SAP News Center. Accessed August 11, 2020. <https://news.sap.com/2020/06/corona-warn-app-deutsche-telekom-sap/>
13. Department of Health and Human Services. *21st Century Cures Act: Interoperability, Information Blocking, and the ONC Health IT Certification Program.*; 2020. <https://www.federalregister.gov/d/2020-07419/>
14. Centers for Medicare & Medicaid Services. *Medicare and Medicaid Programs; Patient Protection and Affordable Care Act; Interoperability and Patient Access for Medicare Advantage Organization and Medicaid Managed Care Plans, State Medicaid Agencies, CHIP Agencies and CHIP Managed Care Entities, Iss.*; 2020. <https://www.federalregister.gov/d/2020-05050/>
15. FHIRPath (Normative Release). Accessed August 11, 2020. <http://hl7.org/fhirpath/>
16. RESTful API - FHIR v4.0.1. Accessed August 11, 2020. <https://www.hl7.org/fhir/http.html>
17. Google Inc. Flutter - Beautiful native apps in record time. Accessed June 9, 2020. <https://flutter.dev/>
18. Keycloak. Accessed August 17, 2020. <https://www.keycloak.org/>
19. Localization Platform for Translating Digital Content | Transifex. Accessed August 17, 2020. <https://www.transifex.com/>
20. HL7/fhirpath.js: Javascript implementation of FHIRPath. Accessed August 11, 2020. <https://github.com/hl7/fhirpath.js/>
21. Mandl KD, Kohane IS. 2009. No Small Change for the Health Information Economy. *N Engl J Med*. 360(13), 1278-81. doi:<https://doi.org/10.1056/NEJMp0900411>. [PubMed](#)