

Article

Not peer-reviewed version

A Digital Vaccination Pass Using Fast Healthcare Interoperability Resources: A Proof of Concept

Tobias Klausen , Valentin Hartig , [Dominik Fuchs](#) , Nils Krueger , Vincent Melchior Jeltsch , Raffael Bild , [Raphael Scheible](#) *

Posted Date: 31 January 2024

doi: 10.20944/preprints202401.2245.v1

Keywords: FHIR; Vaccination Pass; Software; Public Health Informatics; Expert Systems; Inference Engine; HL7; Medical Data Interoperability; Prototype; Feasibility Analysis



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

A Digital Vaccination Pass Using Fast Healthcare Interoperability Resources: A Proof of Concept

Tobias Klausen ^{1,†}, Valentin Hartig ^{1,†}, Dominik Fuchs ¹, Nils Krueger ^{1,2}, Vincent Jeltsch ², Raffael Bild ³ and Raphael Scheible ^{3,*}

¹ School of Computation, Information and Technology, Technical University Munich, Munich, Germany

² Avi Medical Operations GmbH, Munich, Germany

³ Institute of Artificial Intelligence and Informatics in Medicine, Chair of Medical Informatics, University Hospital rechts der Isar, Technical University of Munich, Munich, Germany

* Correspondence: raphael.scheible@tum.de; Tel.: +49-(0)89-4140 4367

† These authors contributed equally to this work.

Abstract: The traditional manual recording of vaccination records in Germany faced challenges during the COVID-19 pandemic, prompting the introduction of a Covid smartphone app with QR codes. However, this solution brought new challenges, emphasizing the need for a centrally managed European digital vaccination record for efficiency and validity. Thus, this study assesses the feasibility of using the HL7 FHIR standard in the healthcare industry for implementing a digital vaccination pass management and monitoring system. The system aims to offer convenience and improved efficiency for both patients and healthcare providers while promoting interoperability with other healthcare systems. We developed a prototype using modern technologies, such as React, Quarkus, and Keycloak, evaluating the applicability of FHIR in promoting interoperability within the healthcare industry. Results indicate potential benefits for patients and healthcare providers, offering secure access to immunization records, personalized recommendations, and streamlined management. However, challenges in integrating nuanced vaccination processes into the standardized FHIR system, necessitating custom extensions, and issues with manual data entry and identity provider integration, pose obstacles. Despite these challenges, the study suggests that implementing HL7 FHIR can enhance efficiency, data accessibility, and accuracy in the vaccination process, supporting broader digitization efforts in the German healthcare system and beyond.

Keywords: FHIR; vaccination pass; software; public health informatics; expert systems; inference engine; hl7; medical data interoperability; prototype; feasibility analysis

1. Introduction

Currently, in Germany, vaccination records are traditionally recorded in a physical yellow booklet that is manually filled out by doctors. This booklet is often transferred between different hospitals and practices as individuals receive vaccinations from various healthcare providers throughout their lives. During the COVID-19 pandemic, the drawbacks of this analog system became visible: In Germany, exit restrictions and various restrictions on social life were imposed, which were gradually eased. Initially, however, these relaxations only applied to those who had been vaccinated. This created opportunities for fraud, with unvaccinated people faking the analog vaccination booklet and therefore potentially putting themselves and others into danger [10] and leading to false statistics regarding breakthrough infections [11]. With the release of the COVID-19 smartphone app CovPass, medical doctors and vaccination centers additionally handed out QR codes, which should increase the validity of the vaccination status. However, alternatively, pharmacists were also enabled and put responsible for issuing these QR codes, using the booklet to check vaccination status and identity. As a result, fakes were still possible, and the entire process led to increased effort for pharmacists and thus to additional costs. Even fake pharmacies were discovered, handing out valid fake certificates and unfortunately, due to the lack of a mutual application all over Europe, this issue was not solved

on a wider range, but solely in the German app [12]. A European digital vaccination record managed only by the vaccinating doctors and centers would have remedied this, saved resources, and led to even greater validity. In addition to the correctness and the locally independent access option via the Internet, the possibility of a reminder function for the necessity of re-vaccinations could be a beneficial feature. Medicine develops gradually and established knowledge can always become outdated as a result. One example of this is the yellow fever vaccination. In 2013, the WHO determined that a single dose of the yellow fever vaccine provides lifelong protection for most individuals. In 2015, the Permanent Vaccination Commission (STIKO) aligned with the WHO's recommendation, suggesting booster shots only for specific groups like pregnant individuals or those with immunodeficiency, such as HIV. However, a 2020/2021 systematic review with meta-analysis revealed that lifelong immunity cannot be assured with just one vaccine dose, and the rate of waning immunity depends on age and initial immunization status [13]. If a vaccination app forwarded such information to the patient, they would potentially make an appointment with the doctor for a booster, instead of relying on outdated knowledge. Generally, making patient data accessible across devices and systems has many benefits, as has been thoroughly researched in science [2–6]. However, the sharing of medical data between hospitals, or even departments of a single hospital, in Germany is often challenging. One of the greatest challenges is building systems that can provide data in an interoperable manner. Health Level 7 (HL7) attempts to solve this problem with its Fast Healthcare Interoperability Resources (FHIR) standard, applied to manifold areas in medical informatics [7–9]. FHIR builds on the previous standards HL7 v2.x and HL7 v3. As HL7 v3 has been perceived as too complex and too expensive to use in practice by the community, FHIR strives to build a more consistent and accessible standard for interoperable exchange of medical data [1]. Although the first release of the FHIR standard dates back to 2014, the normative version of FHIR has been released on October 30, 2019. Since then, it has only been partly adopted in industry, as many of the existing systems still rely on HL7 v2 or have not introduced digital data exchange standards at all. To this date, no digital vaccination pass was developed, nor any system show-casing a prototypical implementation was released. Also, no FHIR profiling existed for such a use case.

Thus, in this study, we investigate the feasibility of using FHIR for implementing a digital vaccination pass. We propose a digital vaccination status management and monitoring system that has the potential to improve the efficiency and effectiveness of the vaccination process for both patients and Medical Doctors (MDs). For patients, the system provides a secure and convenient way to examine their immunization records, allowing them to easily check their vaccination status. The system also offers personalized recommendations for missing vaccinations based on individual factors such as age, medical history, and current immunization status. For MDs, the system offers a more efficient and secure way to manage vaccination records, reducing the need for manual record-keeping and minimizing the risk of errors. It also enables doctors to easily access a patient's complete vaccination history, allowing them to make more informed decisions while streamlining the vaccination process through increased efficiency in administration and follow-up. The use of FHIR as the underlying standard in the proposed system enables interoperability with a wide variety of other healthcare systems, facilitating the exchange of vaccination information and supporting the broader goal of digitizing the healthcare system. The proposed system has been prototyped using modern standard technologies, such as React [14], Quarkus [15] and Keycloak [16].

1.1. Fast Healthcare Interoperability Resources

Fast Healthcare Interoperability Resources (FHIR) is a data exchange standard for the healthcare industry, developed by HL7. It is designed to enable the exchange of EHR between different systems and organizations. For this, it utilizes modern web standards, including Hypertext Transfer Protocol (HTTP), JavaScript Object Notation (JSON), and Extensible Markup Language (XML) to provide a common language for health data, allowing for seamless communication and data sharing.

A key web standard used in FHIR is Representational State Transfer (REST), which provides a set of architectural principles for web-based systems. REST allows for the creation of a standardized, modular Application Programming Interface (API) that can be easily accessed by different systems.

This enables the creation of scalable, flexible systems that can be easily integrated with other applications.

FHIR is composed of a set of modular components, known as "resources," that can be combined to create a wide range of clinical and administrative data models. This modular design allows for a high degree of flexibility and customization, making FHIR well-suited for a wide range of use cases.

By using FHIR, MIS can more easily share and access data, which can help improve the quality and accuracy of the information that is available to healthcare providers. This, in turn, can help improve patient care and make the administrative aspects of running a medical practice or healthcare facility more efficient. Additionally, because FHIR is a standardized format, it can help to ensure that the data being exchanged is structured and consistent, which can make it easier to use and analyze. This is particularly useful for medical documentation and billing, as it helps to ensure that the information being recorded is accurate and complete.

1.2. FHIR Implementation Guides

FHIR Implementation Guides are sets of instructions and rules for implementing the FHIR standard in a specific healthcare context [17]. These guides are designed to help healthcare organizations to adopt the FHIR standard and to improve the interoperability of their systems.

FHIR Implementation Guides typically include information on the specific data elements and resources that need to be exchanged, as well as any rules that need to be followed. At the core of this are resource profiles that both restrict and extend base resources defined by the core FHIR standard. Constraints placed on resources can for instance include the cardinality of their attributes, making some of them required or specifying a precise number of occurrences. Extensions are used to add custom attributes of arbitrary complexity to resources. An Implementation Guide may also include custom and standard terminologies, such as the International Statistical Classification of Diseases and Related Health Problems - Version 10 (ICD-10) [18], as well as rules for how they are used in resources.

The SUSHI software tool [19] enables the creation of FHIR Implementation Guides using FHIR Shorthand [20] text files to define resource profiles. It provides a means of efficiently and accurately constructing FHIR Implementation Guides, enabling the standardized representation of healthcare data and information exchange between systems.

1.3. FHIR RESTful API

The FHIR RESTful API is a web-based interface that allows applications to access and exchange healthcare data using FHIR Resources and the REST interface architecture style [6]. It supports search and discovery operations, allowing clients to search for specific resources using a query language specified by the FHIR specification. The search parameters include elements such as the resource type, the resource identifier, and search criteria based on the data elements in the resource. For example, a client may use a search operation to find all patient records with a specific diagnosis code or to retrieve a specific lab result by its identifier. The server responds with a collection of resources that match the search criteria, along with relevant metadata such as the total number of matching resources and links for pagination. More elaborate search queries can be formulated using FHIR Path.

1.4. FHIR Path

FHIR Path is a language for querying and manipulating data in the FHIR data model [21]. It is similar to other query languages, such as XPath [22] and SQL [23], but is specifically designed to work with the hierarchical structure of FHIR resources. FHIR Path allows users to extract and manipulate specific pieces of information from these resources in a standardized and interoperable manner. It uses a combination of elements and operators to perform operations on FHIR data. Elements are used to identify specific data items within a FHIR resource, and operators are used to perform operations such as filtering, sorting, and aggregation on the data. For example, a FHIR Path query might use the

"patient" element to identify a specific patient record and the "where" operator to filter the records based on specific criteria, such as the patient's age.

2. Materials and Methods

The development of the vaccination pass showcase application involved a multi-faceted approach, including the creation of a domain model to represent the problem space, the representation of this domain model using FHIR resources, and the selection and implementation of both existing open-source software and new software components to meet the requirements of the application. In the following sections, we detail the specific methods and approaches employed in each of these areas.

2.1. Domain Model

To develop a domain model that accurately reflects the real-world requirements of vaccination management, we utilized both the vaccination recommendations for Diphtheria, Tetanus, Polio, and SARS-CoV-2 from the Robert-Koch-Institute (RKI) [24] and the product information for the corresponding recommended vaccines as examples to be represented in the model [25]. To maintain conformance to the FHIR standard, we adapted already existing FHIR base resources to our domain whenever a conceptually matching resource existed. Otherwise, we used the "Basic" FHIR resource and extended it accordingly. Subsequently, a FHIR Implementation Guide was created.

2.2. System Architecture

The stated goals were realized through a systematic process. First, we identified system components at an abstract level. Next, we aligned the requirements for each component with an appropriate existing software solution or developed a custom implementation specifically for the vaccination pass use case. The following system components were identified, and their relationships are displayed in Figure 1:

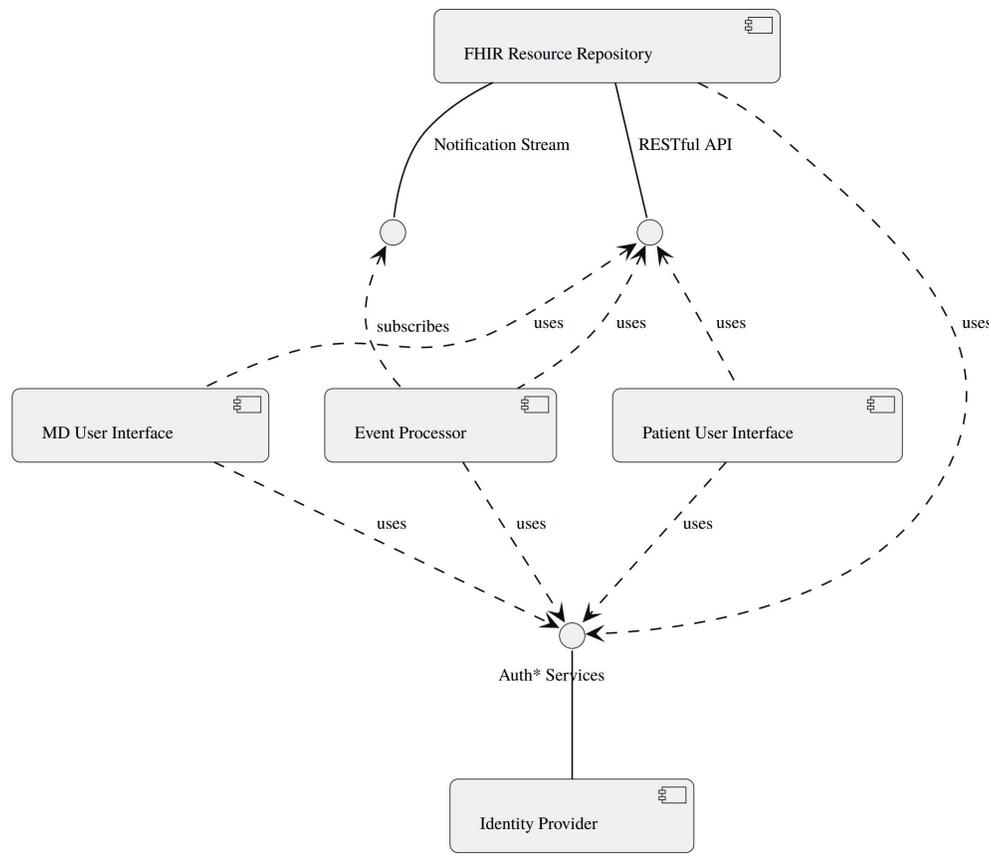


Figure 1. Architecture Diagram of the defined system components.

2.3. FHIR Path

- **FHIR Resource Repository** to store, modify, access and search both user data (i.e. personal immunizations) and domain knowledge (i.e. vaccination recommendations by RKI) as FHIR resources.
- **Event Processor** that automatically generates personalized vaccination recommendations when user data or domain knowledge changes.
- **Patient User Interface** to visualize an individual's immunization record, present recommendations to them, and display the modeled domain knowledge on vaccines and targeted diseases.
- **Medical Doctor User Interface** to allow a practitioner to view the immunization record for each patient in the system, record new immunizations, generate and adjust recommendations, and view and update domain knowledge on vaccines and targeted diseases.
- **Identity Provider** that allows the patient and MD accounts to be managed and serves as a centralized authority on user roles and permissions for the other components.

The requirements for the system components are described in further detail in the following subsections.

2.3.1. FHIR Resource Repository

The FHIR resource repository serves as a centralized store for all FHIR resources consumed and created in the system. It is the only component that can persist FHIR resources. Components that create resources transmit them to the FHIR resource repository to save them. Components that require existing FHIR resources query it to retrieve them. To fulfill this role, the software component should implement the FHIR RESTful API to allow other software components to query and create

FHIR resources. Additionally, all incoming FHIR resources should be validated to ensure the correctness of saved resources including conformance to custom profiles. Supporting the event processor component requires the repository to implement a method for subscribing to changes of the FHIR resources. In addition, integration with the identity provider component requires the FHIR resource repository to implement authentication of clients that is based on an external system as an authority on user accounts and roles.

2.3.2. Event Processor

Generating and updating immunization recommendations based on a patient's immunization record and risk profile is handled by the event processor component. Software that fulfills the requirements of this component needs to re-evaluate immunization recommendations of all affected patients if one of the following situations occur:

- a patient is created;
- an immunization is added to the patient immunization record;
- recommendations for which patients should receive vaccines targeting a disease are changed;
- a patient plans a vacation for which additional immunities are recommended.

This component must be able to subscribe to the feed of FHIR resource changes offered by the FHIR resource repository. It should use the FHIR RESTful API to retrieve additional information required for the re-evaluation triggered by a FHIR resource change and to store updated immunization recommendations in the repository.

2.3.3. Identity Provider

The identity provider is the central store for the patient and MD user accounts that provides information on user authentication and roles to all other components. To fulfill the requirements of this component, software should offer a UI that allows both patients and MDs to log in to their accounts. On a successful user authentication in the log-in interface, the identity provider should issue a token that can be used to prove the identity of the user in the communication with other components. This way, the log-in interface of the identity provider can be integrated with the patient and MD UIs. These requirements can be fulfilled by implementing the industry standard OpenID Connect (OIDC) [26].

2.4. User Interface

2.4.1. Identity Provider

The Patient User Interface provides patients with information regarding their current immunization status, as well as their immunization history. By employing an iterative prototyping approach, beginning with the creation of low-fidelity User Interface (UI) mock-ups, it was identified that a mobile UI would be most appropriate for the Patient UI. The creation of high fidelity mock-ups in Figma [27] and their subsequent evaluation led to the conclusion that a minimal UI design with color-based micro interactions would lead to a better user experience, as important parts of the application can be identified easier. To further enhance the usability of the Patient UI, qualitative interviews with experts (doctors) were conducted to understand the needs and preferences of patients. These interviews helped to identify a minimal set of features that the interface should include, based on the doctors' knowledge of the current use of paper vaccination cards.

2.4.2. Medical Doctor-facing

The Medical Doctor User Interface aids MDs in providing immunization recommendations to patients. It should be coherent with the Patient UI and leverage common design elements and functionalities. The UI should enable MDs to get insights into a patient's vaccination history and provide an overview of the patient's current immunization status. Furthermore, it should enable the MD to make informed decisions about the next steps in a patient's immunization based on generated

recommendations. The MD should be able to configure the patient-facing application via the UI and configure the recommendation engine through custom settings for each disease. All design decisions, represented in high fidelity UI models designed in Figma, are based on interviews with MDs. To ensure the usability and effectiveness of the MD UI in practice settings, it should be optimized for desktop use. This design choice was made with the specific needs and usage patterns of MDs in mind and represents a departure from the design considerations for the Patient UI.

2.4.3. Implementation

Initially, the MD UI was a standalone application. However, to optimize reuse in this prototypical implementation, the MD UI and the Patient UI were integrated into a single client-facing application, resulting in a reduction of boilerplate code for API calls, among other things. Both are realized as a web application based on the JavaScript UI framework React. The UIs utilize ChakraUI, a component library that provides a wide range of pre-designed UI elements [28], in combination with custom CSS generated using Sassy Cascading Style Sheets (SCSS).

Both UIs utilize subscriptions to receive create and update notifications from the FHIR server. Upon initial loading, the UIs initialize themselves through the utilization of the FHIR REST API. Upon receipt of update-notifications from the FHIR server, the UIs evaluate if the update may impact the data utilized within the view, and if so, fetch the relevant data again.

3. Results

The following section outlines a comprehensive explanation of the developed prototype, including a thorough demonstration of how the requirements previously specified were satisfied through the integration of various open-source and custom developed software components.

3.1. Domain Model

To evaluate an individual patient's immunization status, the vaccination pass application requires personal data from patients as well as general domain knowledge about vaccination recommendations, diseases, vaccines, and the vaccination process. Figure 2 shows the resulting domain model using the Unified Modeling Language (UML).

All patient-specific data is represented by the resources "Patient", "Immunization", "Immunization-Recommendation" and "VacationPlan". "Patient" represents all general information such as name, birth date, and address. The address is of particular importance as additional immunizations based on location may be recommended. An instance of an "Immunization" models that the patient received a single dose of vaccine. It corresponds to a single row in a physical vaccination booklet. An "ImmunizationRecommendation" is a single vaccination dose that should be administered in the future. The "VacationPlan" allows the patient to add vacation destinations so that applicable travel vaccinations can be recommended. Specifying the departure date ensures that immunization is achieved before departure if multiple shots are required.

All domain knowledge is represented by "TargetDisease", "PopulationRecommendation", "Medication", "VaccinationScheme", and "VaccinationDose". A "TargetDisease" contains general information about a disease (i.e. Tetanus) such as its name and ICD-10 code. A "PopulationRecommendation" predicates that all persons of the specified age living in the specified location should have immunization against a certain target disease. An instance of "Medication" provides information on a single vaccine such as the diseases it protects against. Note that one vaccine can protect against multiple diseases, as is the case with Boostrix, which offers protection from Diphtheria, Tetanus, and Pertussis. A "VaccinationScheme" represents a specific vaccination scheme of the referenced vaccine, for example, a fast or booster scheme. It consists of one or more instances of "VaccinationDose", representing either a single shot ("VaccinationDoseSingle") or infinitely repeated shots ("VaccinationDoseRepeating"). For example, a scheme of the Boostrix vaccine could consist of five "VaccinationDoseSingle" instances at the ages of 2 months, 4 months, 11 months, 5-6

years, and 9-16 years as well as one "VaccinationDoseRepeating" instance which is administered every 10 years after the last single dose.

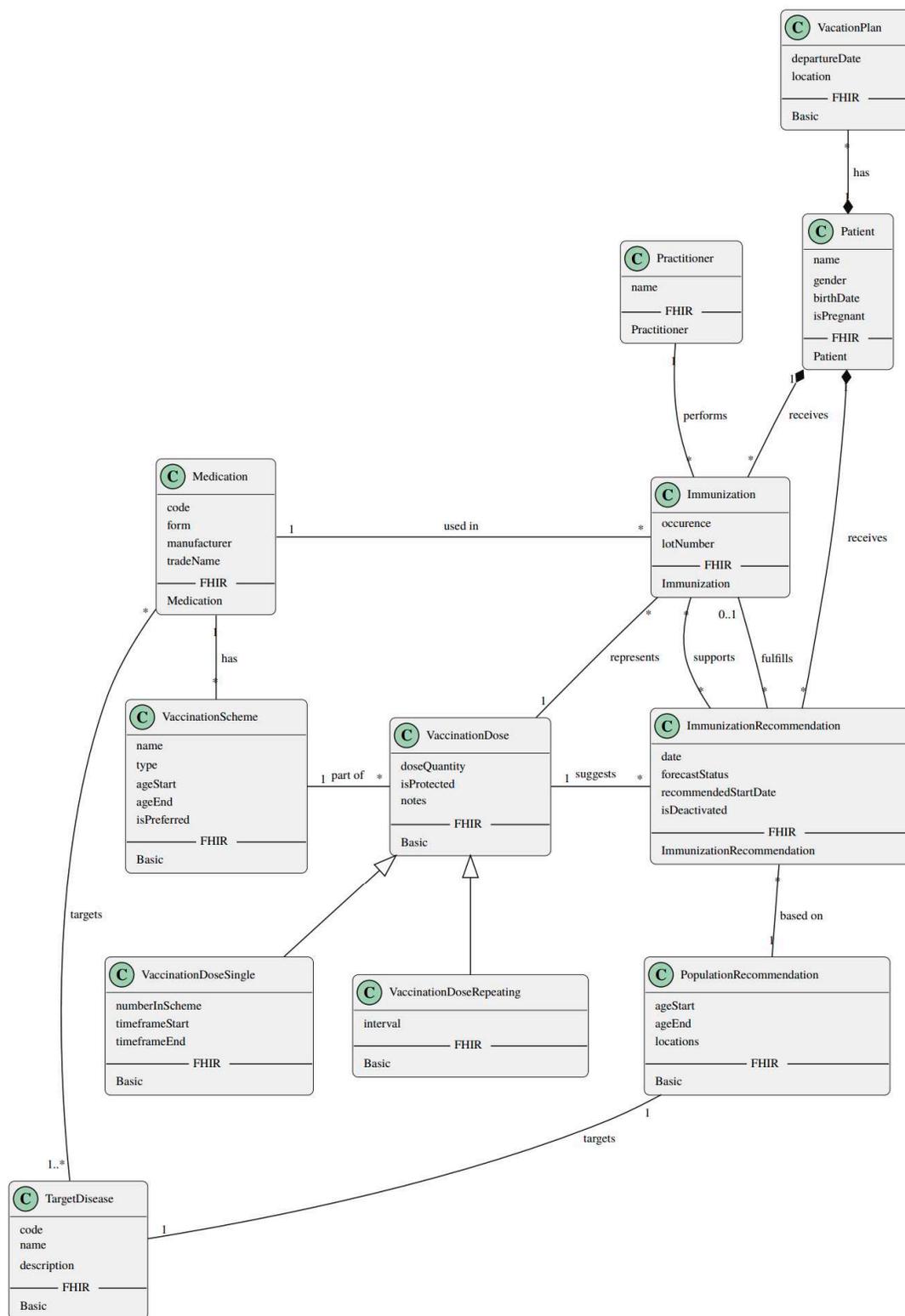


Figure 2. Domain Model identified during the evaluation of immunization recommendations and manufacturer product information of vaccines.

3.2. FHIR Implementation Guide

A FHIR Implementation Guide was created to adapt the FHIR standard to our needs according to the domain model specified in Figure 2. For this purpose, the domain-specific language FHIR Shorthand (FSH) [20] as well as its compiler SUSHI [19] were used.

For each class in the domain model, a FHIR base resource was identified and customized using profiles. This included describing constraints on the use of elements of the base resource as well as defining extensions that contain additional attributes not included in the base resource. For example, in our version of the resource "Immunization", one vaccine code must be a Pharmazentralnummer (PZN), a German national drug code. In addition, an extension was defined that references a "VaccinationDose".

Furthermore, search parameters for extensions were defined. This allows filtering resources based on attributes of extensions. For base attributes, the most common search parameters are already defined as part of the base resource definition.

To specify the vaccination scheme type, a code system was created. It determines which types exist and how they are understood.

Using the HL7 Implementation Guide Publisher [29], a full HTML-based documentation of the Implementation Guide has been created [30]. Figure 3 depicts an example rendering of the constraints contained in the implementation guide.

The screenshot shows a web interface with four tabs: 'Text Summary', 'Differential Table', 'Snapshot Table', and 'All'. The 'Snapshot Table' tab is active. Below the tabs is a table with the following columns: Name, Flags, Card., Type, and Description & Constraints. The table lists the following elements:

Name	Flags	Card.	Type	Description & Constraints
Extension		0..*	Extension	VaccinationSchemeExtension
id		0..1	string	Unique id for inter-element referencing
Slices for extension		3..*	Extension	Extension Slice: Unordered, Open by value:url
extension:name	S	1..1	Extension	Extension
extension:type	S	1..1	Extension	Extension
extension:isPreferred	S	1..1	Extension	Extension
extension:ageStart	S	0..1	Extension	Extension
extension:ageEnd	S	0..1	Extension	Extension
url		1..1	uri	"https://fhir-vaccination-pass.github.io/fhir-implementation-guide/StructureDefinition/vp-vaccination-scheme-extension"

At the bottom of the table, there is a link: [? Documentation for this format](#)

Figure 3. Rendering of the constraints placed on the "VaccinationSchemeExtension" FHIR extension used to model vaccination schemes.

3.3. FHIR Resource Repository

The LinuxForHealth FHIR Server (formerly IBM FHIR Server) [31] was used as the implementation of the FHIR resource repository in our prototype. This FHIR server was chosen because of the following features:

- FHIR RESTful interface version R4;
- delegation of authentication and authorization to OIDC-compliant identity provider;
- event subscriptions for changes to FHIR resources;
- FHIR resource validation with custom implementation guide.

The FHIR server's REST request authentication was configured to authenticate requests made to its API by checking for the presence of a JSON Web Token (JWT) in the request headers and validating the JWT token using the public key published in the OpenID Connect (OIDC) endpoint of the prototype's Keycloak instance. This way, the server verifies that requests made to it originate from users that have successfully obtained JWT tokens issued by Keycloak.

To support event subscriptions the server's notification service was enabled and configured to provide FHIR resource change events as messages on a Web-Socket endpoint.

To provide validation of FHIR resources according to the prototype's Implementation Guide, a Java JAR file containing an implementation of the FHIRRegistryResourceProvider interface must be created and placed in the FHIR server's userlib directory. This was achieved by making use of the PackageRegistryResourceProvider class provided by LinuxForHealth and packaging the prototype's Implementation Guide as a FHIR package. This process was automated as part of the tooling for the creation of the prototype's Implementation Guide.

3.4. Identity Provider

The event processor component was implemented as a stateless Java microservice based on the Quarkus framework. The event processor connects to the FHIR server's websocket endpoint to receive events for each modification of a FHIR resource. Events are routed internally to the corresponding function that handles them by resource type and operation type, for instance Patient/create. Resources modeled with the "Basic" resource type due to them representing a concept missing in the FHIR standard present an additional challenge in this process: They are additionally routed by utilizing the FHIR profile URL given in the metadata of the resource. This way, a resource of type "Basic" that represents information about a target disease can be routed to one handling function, while a "Basic" resource that represents information about a vacation plan can be routed to another handling function. Resources that do not conform to the prototype's implementation guide are simply ignored by the event processor, regardless of the resource type.

Once routing is completed, the called event handling function will create or update immunization recommendations for one or multiple patients. Since the specific function handling the Patient/create event is the most complex (and since all other functions can be trivially replaced by deleting all immunization recommendations and generating them from scratch for every patient), we describe it in the following:

1. Identify all diseases the patient should have immunization coverage for, considering the patient's age, places of residence, and vacation plans.
2. Derive the actual coverage from existing immunizations and match them to the diseases identified in step 1.
3. Generate immunization recommendations for diseases without existing immunizations:
 - a. Select vaccines that exactly cover all diseases without existing immunizations. Prefer combination vaccines to reduce the number of doses.
 - b. Choose vaccination schemes for all selected vaccines. Ensure that immunization is completed before traveling. Prefer schemes with type "standard" and priority "preferred".
4. Generate immunization recommendations for diseases with existing immunizations: Recommend the next dose for every administered vaccine in the current scheme. Change the scheme to type "booster" if all doses of type "standard" or "fast" have been administered.

3.5. Identity Provider

Due to the industry standard nature of OIDC, multiple commercial and non-commercial software products exist that implement it. In the vaccination pass prototype Keycloak was selected as the identity provider component, as it is open-source software, and the project provides Docker

[32] containers of the application that make it easy to self-host for prototyping purposes. Note that this component is rather interchangeable, a production deployment institution should make use of existing identity infrastructure if it implements OIDC. If desired, it is also possible to utilize existing identity infrastructure to log in to a user account store in Keycloak, if replacing Keycloak is not desirable.

In the prototype, the web application that provides the patient and MD user interfaces, the event processor and the LinuxForHealth FHIR server have been added as clients in the Keycloak configuration to allow them to initiate login flows, validate authentication tokens, and retrieve information about user accounts.

3.6. User Interfaces

In the following we present the two user interfaces: the administrative UI for the MD and the consumer interface for the patient. A demonstration video of the application can be found in Supplementary Materials.

3.6.1. Patient-Facing

To realize the proposed design of the patient-facing UI, a web application was developed. The color-coded micro interactions previously mentioned were included in the design to highlight important elements of the UI and draw attention to alerts, notifications, and other important information. It intends to quickly convey the status or importance of the information being presented to the user. The micro-interactions are divided into four states, which are distinguished by their color coding:

- The Default State is displayed when there is no available information about an individual's immunization status for a particular disease, or when immunization against the disease is optional. This state is represented by the default color used in the UI.
- The Immunized State indicates that an individual has completed all necessary immunizations and no further action is required at this time. This state is represented by the color green to convey the absence of any outstanding issues.
- The Due State indicates that an individual's immunization for a particular disease is imminent or will be necessary for the near future, and that action should be taken to receive it. This state is represented by the color orange to signify its urgency.
- The Overdue State refers to a situation in which an individual's immunity to a particular disease has waned or never existed, and it is recommended that they receive immunization against the disease. This state is depicted using the color red to indicate its importance.

The Patient UI consists of three components that allow patients to view their immunization information from different perspectives, depending on their current needs:

Recommendation Dashboard: The Patient UI includes a recommendation dashboard, which is displayed in Figure 4, that provides a comprehensive overview of a patient's vaccination status for various diseases and displays vaccinations that are pending, due, or overdue. This feature aims to facilitate the patient's ability to access clear, concise information about recommended vaccination measures. The recommendation dashboard calculates an aggregated vaccination status, which is a summary of the patient's vaccination status across multiple diseases. This is achieved by considering the vaccination information for all relevant diseases. The aggregated vaccination status allows the patient to easily ascertain their overall vaccination status, rather than having to review the vaccination status for each disease individually.

Vaccination History: The vaccination history shown in Figure 5a is a comprehensive record of all immunizations administered to an individual, organized by the specific disease and the corresponding date of vaccination. This representation allows the patient to retrospectively examine their vaccination status for various diseases and the temporal relationships between them.

Patient Disease Record: The patient disease record, displayed in Figure 5b, serves as the central repository for information related to immunizations within the system. This record lists all diseases for which immunization is recommended or required and includes a summary of the patient's

vaccination status for each disease. When a specific disease is selected, a detailed overview of relevant information is provided, including general information about the disease (e.g., population recommendations, available vaccines, countries where the disease is prevalent), as well as personalized information regarding the patient's vaccination status for that disease. The header of the detailed overview includes micro-interactions that provide a summary of the patient's overall vaccination status and can be expanded to display information about upcoming immunizations and previous vaccination procedures, complete with details about each individual vaccination. The detail overview is depicted in Figure 5c.

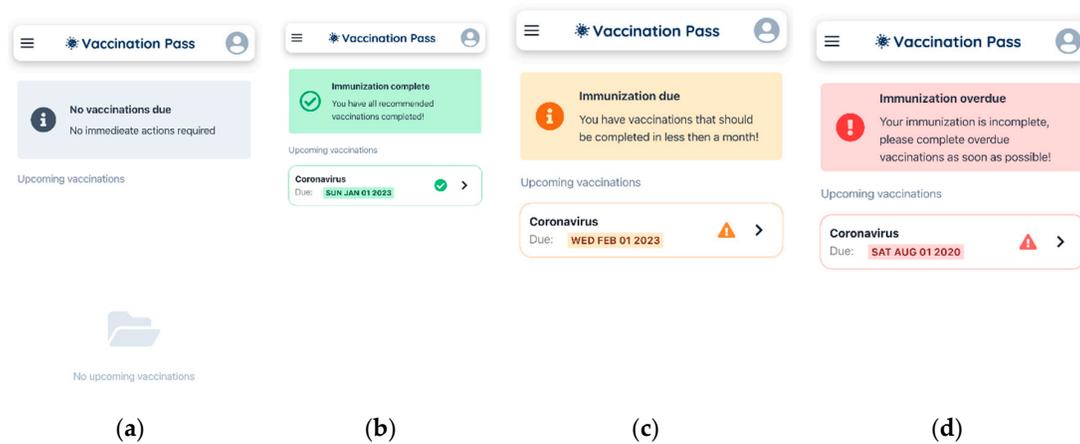


Figure 4. Recommendation Dashboard in the Patient UI showcasing the micro-interaction states: (a) default state, (b) immunized state, (c) due state and (d) overdue state.

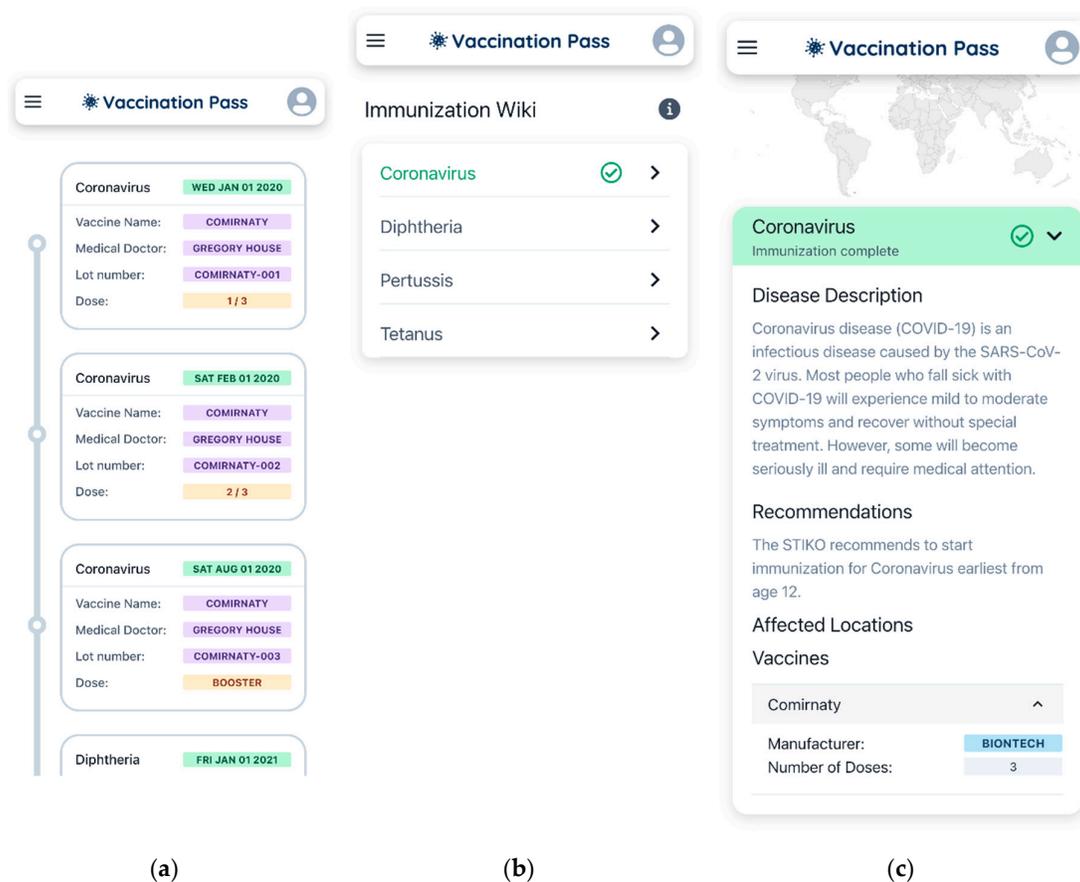


Figure 5. Various views in the Patient UI: (a) vaccination history (b) disease record overview and (c) detailed disease record.

3.6.2. Medical Doctor-Facing

The MD UI has been implemented based on the high-fidelity mockups created with Figma. This implementation grants the MD the ability to configure the system and manage patients effectively. The interface is structured into various components to facilitate the smooth navigation and utilization of various features and functions. The MD UI consists of the following components:

Base Configuration: Via the Base Configuration interface, the MD can configure various aspects of the system. This interface enables MDs to specify the content that is displayed in the patient UI, as well as the relationships between different entities such as diseases and vaccines. To manage diseases, MDs can use Create, Read, Update and Delete (CRUD) operations. For example, they can create a target disease with a name, ICD-10 code, and a description. They can also create population recommendations for a particular disease, specifying the affected ages and locations using ISO-3166 location codes [33]. Similarly, MDs can use CRUD operations to manage vaccines as shown in Figure 6. They can create a vaccine with a name, PZN code, manufacturer, and target disease. They can also create vaccination schemes for a particular vaccine, specifying the name, age range, and type of the scheme. In addition, they can create doses for a vaccination scheme, specifying the dose number and quantity. MDs can also create booster vaccination schemes that repeat indefinitely.

MD Patient Dashboard: MDs can manage individual patients using the patient dashboard displayed in Figure 7. This dashboard allows MDs to select patients from a list or search for a particular patient. The dashboard provides a quick overview of a patient's immunization status using different widgets. These widgets summarize important information about the patient, such as their name and date of birth, as well as all recommendations for immunizations. The widgets also display the most recent vaccinations and a list of the most important diseases that a person should be immunized against.

MD Patient Vaccination History: The MD patient vaccination history provides an overview of a patient's past immunizations and is based on the patient's vaccination history in the patient UI. The MD patient vaccination history serves to provide MDs with knowledge on when to administer vaccines to patients based on their previous immunization history. This can help to ensure that patients receive the appropriate vaccines at the appropriate intervals, ultimately improving vaccination rates and public health outcomes.

MD Patient Disease Record: The MD patient disease record is a feature that provides an overview of all diseases that are recorded in the system and whether a patient is immune to them. When a disease is clicked, detailed information about the patient's vaccination against that disease is displayed, including recommendations and past immunizations. This helps MDs to understand a patient's immunity status for various diseases and make informed decisions about vaccinations.

The screenshot shows the 'Vaccination Pass' configuration interface. At the top, there is a 'Patient Search' field. The main area is titled 'Vaccine Information' and includes a 'Save Changes' button. The configuration fields are as follows:

- Manufacturer:** Biontech
- PZN Code:** Boostrix-PZN
- Form:** Injection
- Target Diseases:** Diphtheria, Pertussis, Tetanus
- Vaccination Scheme:**
 - Base Immunization:** Standard Scheme (checked)
 - Recommended from age:** 4
 - Recommended until age:** (empty)

A table lists the doses for the scheme:

DOSE NUMBER	TYPE	DOSE QUANTITY	TIME FRAME	NOTES
1 / 3	Single	0.5	Immediately	Note x
3 / 3	Single	0.5	After 150 to 160 days	Note x
2 / 3	Single	0.5	After 30 to 40 days	Note x

Below the table is an 'Add Dose' button and a 'Booster' dropdown menu. At the bottom, there is an 'Add Scheme' button.

Figure 6. Base Configuration of vaccines in the MD UI.

The screenshot shows the 'Patient Dashboard' for Hans Müller. The interface includes a sidebar with navigation options: Patient Status, Vaccination History, and Disease Record. The main content area is divided into several sections:

- General Information:**
 - Given name:** Hans
 - Surname:** Müller
 - Birthdate:** 01.01.2000
 - Gender:** Male
 - Country:** Germany
 - State:** Bavaria
- Recommendations:**
 - Coronavirus:** DOSE: BOOSTER (Details)
- Important Diseases:**
 - Coronavirus:** overdue (Details)
 - Diphtheria:** default (Details)
 - Pertussis:** default (Details)
- Recent Immunizations:**
 - Boostrix:** DOSE: 3 / 3 (1.7.2021)
 - Boostrix:** DOSE: 2 / 3 (1.2.2021)
 - Boostrix:** DOSE: 1 / 3 (1.1.2021)

At the bottom, there are buttons for 'Open Disease Record' and 'Open Vaccination History'.

Figure 7. Patient Dashboard in the MD UI.

4. Discussion

4.1. Principal Finding

The chosen scenario for evaluating the maturity of FHIR in the industry is well-suited due to its emphasis on interoperability, as vaccination data needs to be examined and complemented by different doctors and institutions over a patient's lifetime. The developed prototype serves as a proof of concept for the utilization of FHIR in healthcare. The prototype not only digitizes the traditional paper-based immunization record of patients but also provides a comprehensive immunization management platform that enhances the interaction between doctors and patients. Through the use of software interfaces that are strictly limited to FHIR resources, the prototype demonstrates a high level of interoperability. Subsequently we highlight some of the challenges encountered during this evaluation.

4.2. Limitations

4.2.1. Digital Representation of the Immunization Record and Vaccination Process

Through the analysis of the scenario's problem space, it was found that certain parts of the vaccination process were challenging to map to the standardized FHIR resources. This is due to FHIR having a specific perspective on the relationships between different types of medical data, which may not always be applicable. This leads to the need for custom extensions to be used. No FHIR resources exist to represent (i) abstract vaccination recommendations for diseases based on the patient's age and location, (ii) which vaccines target a given disease, and (iii) vaccination schemes modeling the interval between individual vaccination doses. Therefore, custom FHIR resources were created by extending the "Basic" resource. Additionally, already existing FHIR resources were extended. This included referencing newly created custom resources as well as adding simple attributes.

4.2.2. Procurement of Data for Digital Representation

The vaccination pass application requires data from patients including general personal data, their current immunization status, and any vacation plans. In addition, domain knowledge about vaccination recommendations, diseases, and vaccines is necessary.

General personal patient data such as name, birth date, and address already exist in PIS and HIS, which makes it easy to integrate these in an industrial setting. To digitize the current immunization status, all previous vaccinations have to be entered manually by the patient or medical staff. Vacation plans can be added by the patient. In an industrial setting, this is feasible.

However, a major problem is the procurement of data regarding vaccines and their vaccination schemes as well as general vaccination recommendations due to the lack of digital sources that accurately model them in a structured form. Currently, this information is only publicly available in unstructured text formats such as PDFs, provided by federal institutions or the vaccine manufacturer. Some closed-source, pay-to-use databases contain some of the information in a structured format, however, none provides a comprehensive, programmatic overview of vaccinations and their administration schemes. This effectively makes any immunization status inference platform infeasible, as the system's knowledge base has to be created and kept up to date manually by a medical professional. We evaluate that a coherent platform that provides vaccine information, structured data on how to administer vaccines, and general recommendations on when to use a vaccine, would be a necessary next step to building efficient and sustainable vaccination management systems.

4.2.3. Inference Platform Enabled by Digital Representation

Once all data is stored in a FHIR resource repository, it is easily accessible via REST interfaces. In particular, the FHIR Search endpoint is convenient for filtering resources based on attribute values directly in the resource repository. The utilization of LinuxForHealth's fhir-client further simplifies

the API's usage and saves the received FHIR resources as objects from LinuxForHealth's FHIR-model. However, it is infeasible to perform inference directly on these FHIR-model objects as they represent the most general version of any FHIR resource. Therefore, to access an attribute of interest, a FHIR Path evaluation may be necessary. To solve this, mappers were implemented for every resource such that all attributes are instantly accessible.

4.2.4. FHIR Ecosystem Maturity

The integration of FHIR into the standard development stacks of industry players is demonstrated to be feasible through the usage of established technologies such as React with Typescript and Quarkus in our prototype. However, the tooling in the FHIR open-source ecosystem presents limitations, including being prone to bugs and ill maintenance. For example, the HL7 Implementation Guide Publisher used in the generation of the project's implementation guide is susceptible to crashes due to unexpected factors such as missing descriptions in resources.

These limitations are further highlighted in existing server implementations, which have been found to have bugs or missing functionalities. For example, the HAPI FHIR Server [34], which serves as a reference implementation for FHIR servers, has been deemed unsuitable for industry scenarios due to its lack of performance and incomplete support for integrating with identity providers for user authentication and authorization.

The LinuxForHealth FHIR server supposedly provides integration with an identity provider for authentication and authorization through its support of SMART App Launch capabilities [35]. However, the Alvearie Keycloak extension is required to use this feature, which as of January 2023 only supports a legacy version of Keycloak that is no longer in development, making the deployment of this extension not advisable in a newly developed software system [36].

Since the release of the prototype, the Blaze FHIR server [37] has been heavily maturing as it is used in several projects within the medical research domain in Germany. Thus, in the future, an evaluation of this technology appears promising due to its performance and support of the community.

4.2.5. Interoperability

Since the prototype uses a FHIR server as its only means of persisting data, all of the resources in the system can be exported and imported into another FHIR server. While this operation preserves all of the information modeled in the prototype, a different FHIR-compliant system will not interpret the data contained in the custom extensions to the FHIR resources. Therefore, the interoperability with other FHIR systems is limited to the data that is modeled in the standardized FHIR resources and the standardized attributes of those resources. Additional information, such as the knowledge modeling of vaccination schedules or the link of an immunization event to the dose it fulfills in the schedule will not be available in a different system.

An export target that would be feasible to implement with little effort is the FHIR profile "MIO Impfpass" standardized by the German association of statutory health insurance physicians, the Kassenärztliche Bundesvereinigung (KBV) [38]. This profile shares the documentation-focused view on modeling immunization events that are taken in the standardized attributes of the "Immunization" FHIR resource. As such, an export of a patient's immunizations that is compliant with the KBV FHIR profile is possible by omitting the custom extensions linking the immunization events to the vaccination schemes.

4.3. Outlook

The current prototype is based on the UI framework ChakraUI within a React based frontend application. Another possibility to quickly create user-friendly frontends is ReactAdmin (RA) [39], which offers many highly abstracted widgets [40]. RA requires a so-called DataProvider to enable its communication to a specific API-technology. Recently, the FHIR DataProvider was published [41], which would enable us to create a RA application on top of the IBM FHIR server. In future, this might

be an option as frontend technology. Further, a verification functionality considering data protection would be an important feature. Using blockchain technology as an approach to improving security and privacy was introduced, that would enhance the system [42].

5. Conclusion

This study aimed to assess the feasibility of using FHIR in the healthcare industry through the examination of a specific use case: The digital vaccination pass. A proposed digital vaccination status management and monitoring system based on FHIR was successfully developed to improve the efficiency and effectiveness of the vaccination process for both patients and MDs. The use of FHIR as the underlying standard enables interoperability with other healthcare systems and supports the digitization of the healthcare system in Germany. However, the study found that a major challenge in the current FHIR ecosystem for this use case is the absence of a standardized method for modeling knowledge about vaccine application. In particular, information about the number and timing of immunizations required for full immunity is currently presented as infographics or free text. To provide MDs with helpful information and patients with information about their immunity status, a standard for modeling vaccination schemes must be established.

The source code for our prototype, as well as the implementation guide for our custom resources, can be found in repositories of the GitHub organization “FHIR-Vaccination-Pass”.

Supplementary Materials: The following supporting information can be downloaded at the website of this paper posted on Preprints.org, Video S1: Demonstration of the digital vaccination pass.

Author Contributions: Conceptualization, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Data curation, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Formal analysis, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Investigation, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Methodology, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Project administration, Valentin Hartig, Nils Krueger, Raffael Bild and Raphael Scheible; Resources, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Software, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Supervision, Vincent Jeltsch, Raffael Bild and Raphael Scheible; Validation, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Visualization, Tobias Klausen, Valentin Hartig, Dominik Fuchs, Nils Krueger, Raffael Bild and Raphael Scheible; Writing – original draft, Tobias Klausen, Valentin Hartig, Dominik Fuchs and Nils Krueger; Writing – review & editing, Raffael Bild and Raphael Scheible.

Funding: This work was supported by the German Ministry for Education and Research grant number 01ZZ1804A, 01KX2121 and 01ZZ2304A.

Data Availability Statement: The source code for the prototype, as well as the implementation guide for the custom resources, are made available in repositories of the GitHub organization “FHIR-Vaccination-Pass” at the following link: <https://github.com/FHIR-Vaccination-Pass>.

Conflicts of Interest: Nils Krueger and Vincent Jeltsch are both employees of the company Avi Medical, with which this project was done in collaboration with. All other authors declare no conflicts of interest.

References

1. Braunstein, M.L. *Health Informatics on FHIR: How HL7's New API Is Transforming Healthcare*; Springer International Publishing: Cham, 2018; ISBN 978-3-319-93413-6.
2. Beard, L.; Schein, R.; Morra, D.; Wilson, K.; Keelan, J. The Challenges in Making Electronic Health Records Accessible to Patients. *J. Am. Med. Inform. Assoc.* **2012**, *19*, 116–120, doi:10.1136/amiajnl-2011-000261.
3. Milea, D.; Azmi, S.; Reginald, P.; Verpillat, P.; Francois, C. A Review of Accessibility of Administrative Healthcare Databases in the Asia-Pacific Region. *J. Mark. Access Health Policy* **2015**, *3*, doi:10.3402/jmahp.v3.28076.
4. Essén, A.; Scandurra, I.; Gerrits, R.; Humphrey, G.; Johansen, M.A.; Kierkegaard, P.; Koskinen, J.; Liaw, S.-T.; Odeh, S.; Ross, P.; et al. Patient Access to Electronic Health Records: Differences across Ten Countries. *Health Policy Technol.* **2018**, *7*, doi:10.1016/j.hlpt.2017.11.003.
5. Alhaqbani, B.; Fidge, C. Access Control Requirements for Processing Electronic Health Records. In Proceedings of the Business Process Management Workshops; Berlin, Heidelberg, 2008.

6. Dagher, G.G.; Mohler, J.; Milojkovic, M.; Marella, P.B. Ancile: Privacy-Preserving Framework for Access Control and Interoperability of Electronic Health Records Using Blockchain Technology. *Sustain. Cities Soc.* **2018**, *39*, doi:10.1016/j.scs.2018.02.014.
7. Lehne, M.; Luijten, S.; Vom Felde Genannt Imbusch, P.; Thun, S. The Use of FHIR in Digital Health - A Review of the Scientific Literature. *Stud. Health Technol. Inform.* **2019**, *267*, 52–58, doi:10.3233/SHTI190805.
8. Pimenta, N.; Chaves, A.; Sousa, R.; Abelha, A.; Peixoto, H. Interoperability of Clinical Data through FHIR: A Review. *Procedia Comput. Sci.* **2023**, *220*, 856–861, doi:10.1016/j.procs.2023.03.115.
9. Vorisek, C.N.; Lehne, M.; Klopfenstein, S.A.I.; Mayer, P.J.; Bartschke, A.; Haese, T.; Thun, S. Fast Healthcare Interoperability Resources (FHIR) for Interoperability in Health Research: Systematic Review. *JMIR Med. Inform.* **2022**, *10*, doi:10.2196/35724.
10. NDR Geimpft oder gefälscht? Gefahr durch falsche Impfpässe Available online: <https://www.ndr.de/fernsehen/sendungen/panorama3/Geimpft-oder-gefaelscht-Gefahr-durch-falsche-Impfpaesse,impfpaesse110.html> (accessed on 15 November 2023).
11. Litz, C. Impfpass-Fälschungen Sorgen Für Falsche Impfdurchbruch-Zahlen Available online: <https://www.eimsbuetteler-nachrichten.de/impfpass-faelschungen-impfdurchbruch-corona-hamburg-eimsbuettel/> (accessed on 15 November 2023).
12. Ballweber, J. Digitale Zertifikate: Staat hat gefälschten Impfnachweisen wenig entgegenzusetzen Available online: <https://netzpolitik.org/2021/digitale-zertifikate-staat-hat-gefaelschten-impfnachweisen-wenig-entgegenzusetzen/> (accessed on 15 November 2023).
13. Kling, K.; Domingo, C.; Bogdan, C.; Duffy, S.; Harder, T.; Howick, J.; Kleijnen, J.; McDermott, K.; Wichmann, O.; Wilder-Smith, A.; et al. Duration of Protection After Vaccination Against Yellow Fever: A Systematic Review and Meta-Analysis. *Clin. Infect. Dis.* **2022**, *75*, 2266–2274, doi:10.1093/cid/ciac580.
14. React – A JavaScript Library for Building User Interfaces Available online: <https://reactjs.org/> (accessed on 29 January 2023).
15. Quarkus - Supersonic Subatomic Java Available online: <https://quarkus.io/> (accessed on 29 January 2023).
16. Keycloak Available online: <https://www.keycloak.org/> (accessed on 29 January 2023).
17. ImplementationGuide - FHIR v4.3.0 Available online: <https://www.hl7.org/fhir/implementationguide.html> (accessed on 30 January 2023).
18. ICD-10 Version:2019 Available online: <https://icd.who.int/browse10/2019/en> (accessed on 29 January 2023).
19. FHIR/Sushi Available online: <https://github.com/FHIR/sushi> (accessed on 29 January 2023).
20. FHIR Shorthand Available online: <http://hl7.org/fhir/uv/shorthand/> (accessed on 29 January 2023).
21. FHIRPath (Normative Release) Available online: <http://hl7.org/fhirpath/N1/> (accessed on 29 January 2023).
22. Clark, J.; DeRose, S. XML Path Language (XPath) Version 1.0 Available online: <http://www.w3.org/TR/1999/REC-xpath-19991116>.
23. Groff, J.R.; Weinberg, P.N. *SQL, the Complete Reference*; 1999; ISBN 978-0-07-211845-2.
24. Koch-Institut, R. Epidemiologisches Bulletin 4/2022. **2022**, *67*, doi:10.25646/9285.3.
25. STIKO Impf-Infos Available online: <https://www.stiko-web-app.de/home/> (accessed on 29 January 2023).
26. OpenID Connect | OpenID Available online: <https://openid.net/connect/> (accessed on 6 October 2022).
27. Figma: The Collaborative Interface Design Tool. Available online: <https://www.figma.com/> (accessed on 29 January 2023).
28. Chakra UI: Simple, Modular and Accessible UI Components for Your React Applications. Available online: <https://chakra-ui.com> (accessed on 29 January 2023).
29. IG Publisher Documentation - FHIR - Confluence Available online: <https://confluence.hl7.org/display/FHIR/IG+Publisher+Documentation> (accessed on 30 January 2023).
30. FHIR Implementation Guide Available online: <https://fhir-vaccination-pass.github.io/fhir-implementation-guide/>.
31. LinuxForHealth/FHIR Available online: <https://github.com/LinuxForHealth/FHIR> (accessed on 4 August 2022).
32. Docker: Accelerated, Containerized Application Development Available online: <https://www.docker.com/> (accessed on 30 January 2023).
33. ISO 3166-1:2020(En), Codes for the Representation of Names of Countries and Their Subdivisions — Part 1: Country Code Available online: <https://www.iso.org/obp/ui/#iso:std:iso:3166:-1:ed-4:v1:en> (accessed on 29 January 2023).
34. HAPI FHIR - The Open Source FHIR API for Java Available online: <http://hl7.org/fhirpath/N1/> (accessed on 29 January 2023).
35. HL7 FHIR - SMART App Launch Available online: <https://hl7.org/fhir/smart-app-launch/> (accessed on 29 January 2023).
36. Keycloak Extensions for FHIR Available online: <https://github.com/Alvearie/keycloak-extensions-for-fhir> (accessed on 29 January 2023).

37. Kiel, A.; Deppenwiese, N.; Kroll, B.; Engels, C.; Ebert, L.; Lablans, M.; Hummel, M. Feasibility Studies with HL7 FHIR® and Clinical Quality Language.; German Medical Science GMS Publishing House, February 26 2021; p. DocAbstr. 284.
38. MIO Impfpass V1.1.0 - SIMPLIFIER.NET Available online: <https://simplifier.net/im1x0> (accessed on 29 January 2023).
39. Strecker, P.; Boeker, M.; Buechner, S.; Scheible, R. Usability Evaluation of a Modern Multilingual MeSH Browser. *Stud. Health Technol. Inform.* **2022**, *295*, 37–40, doi:10.3233/SHTI220653.
40. Scheible, R.; Strecker, P.; Yazijy, S.; Thomczyk, F.; Talpa, R.; Puhl, A.; Boeker, M. A Multilingual Browser Platform for Medical Subject Headings. *Stud. Health Technol. Inform.* **2022**, *289*, 384–387, doi:10.3233/SHTI210939.
41. Scheible, R.; Alkier, D.; Wendroth, J.; Mayer, J.; Boeker, M. FHIR DataProvider for ReactAdmin: Leveraging User Interface Creation for Medical Web Applications. *Stud. Health Technol. Inform.* **2023**, *305*, 110–114, doi:10.3233/SHTI230436.
42. Ait Bennacer, S.; Aaroud, A.; Sabiri, K.; Rguibi, M.A.; Cherradi, B. Design and Implementation of a New Blockchain-Based Digital Health Passport: A Moroccan Case Study. *Inform. Med. Unlocked* **2022**, *35*, 101125, doi:10.1016/j.imu.2022.101125.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.