


# Towards an effective framework for integrating patient-reported outcomes in electronic health records

Digital Health  
Volume 8: 1–14  
© The Author(s) 2022  
Article reuse guidelines:  
sagepub.com/journals-permissions  
DOI: 10.1177/20552076221112152  
journals.sagepub.com/home/dhj  


Panzhang Wang\* , Tao Li\*, Lei Yu\*, Liang Zhou and Tao Yan

## Abstract

**Background:** In the past decade, electronic modalities are increasingly deployed to integrate patient-reported outcomes into electronic health records. Most popularly, patient portals are used for remote questionnaires, and tablets are provided to patients in-office in case they need help. They are both useful. But some barriers are still in the way, which place burdens on patients and clinicians in the process of routine data collection.

**Objective:** This study aims to describe a portable and scalable framework which can simplify the patient-reported outcome integration by mitigating the related burdens.

**Methods:** A framework was proposed to use a modular approach to replace the tethered approach. The framework was open-sourced on GitHub. After development and testing, it was evaluated on an instrument with 24 questions in a real clinical setting. Patients were randomly selected in every modality-based group. For objective analysis, completion time and response rate were collected. No-show data was collected and analyzed. For subjective analysis, the NASA Task Load Index was used to measure workload, and the Net Promoter Score was used to assess user satisfaction.

**Results:** The model could contain 46,656 questions. A quick response code could store 1120 encoded items. For remote visits, the response rate was improved compared to the portal group (76.6% vs. 61.1%). The completion time was reduced by 37.5% when compared to the tablet group and was reduced by 43.4% when compared to the portal group. The workload for clinicians and patients was both reduced significantly ( $p < 0.001$ ). A higher Net Promoter Score was rated by both clinicians (89.3%) and patients (86.5%). Compared to the portal group, the no-show rate was reduced (11.7% vs. 8.6%).

**Conclusions:** Collecting patient-reported outcomes over a quick response code appears to be an alternative modality to enable a simplified integration. This study provides new insights to collect patient-reported outcomes with interoperability and substitutability in mind.

## Keywords

Electronic health records, patient-reported outcome, privacy, security, quick response code, substitutability, interoperability, integration, data collection method

Submission date: 9 November 2021; Acceptance date: 21 June 2022

## Introduction

### Background

Over the past decade, patient-reported outcomes (PROs) have become a gold standard to hear patients' voice in the routine clinical care,<sup>1–4</sup> which is a burgeoning great interest in integrating the PRO data into electronic health records (EHRs).<sup>5–7</sup> According to the present FDA

Shanghai Jiao Tong University Affiliated Sixth People's Hospital, Shanghai, China

\*Panzhang Wang, Tao Li, and Lei Yu are equally important as the first authors for contribution.

#### Corresponding author:

Panzhang Wang, Shanghai Jiao Tong University Affiliated Sixth People's Hospital, Shanghai, China.  
Email: pz\_wang@hotmail.com



guidance, PRO stands for any report of the status of a patient's health condition that comes directly from the patient, without interpretation of the patient's response by a clinician or anyone else.<sup>8</sup> PRO measures (PROMs) are validated questionnaires for patients to measure PROs.<sup>9</sup> In 2017, the Patient-Centered Outcomes Research Institute (PCORI) developed a Users' Guide to Integrating PROs in EHSs.<sup>10</sup> The guide suggests that an integration system should be accessible for both patients and clinicians. A frontend for patients (FFP) is required to help patients administer questionnaires remotely. A frontend for clinicians (FFC) is also needed in the clinical area. Clinicians could use it to supervise an in-office assessment.

Typically, FFP is implemented as a part of the patient portal. When a remote questionnaire is initiated, patients are first primed about a reminder from the portal. It contains a link that will direct patients to a questionnaire page. Before that, patients might go through some additional steps, such as log-in, password retrieving, and navigation. To avoid disruption of workflows on the day of the clinic visit, the questionnaire is often required to be completed prior to the visit. On the contrary, FFC is often tethered to an EHR, which helps clinicians to manage scheduled visits and PRO requests. With the clinicians' help, patients are able to fill out a form directly. For example, for a walk-in, an assistant will dispatch a PRO request onto a tablet based on the scheduled information. Then, the assistant opens the survey form and passes it to the patient. After the patient submits the form, the assistant will retrieve the tablet and get the patient ready for the encounter.

Due to some technical barriers and security concerns, patient engagement via portals is facing challenges.<sup>11–14</sup> Portz<sup>15</sup> conducted a qualitative descriptive study and found that simplified log-ins and improved navigation could result in a better engagement by mitigating some burdens. Although tablets were helpful for patient engagement by removing log-ins and navigation, Musci et al. found that it is hard to scale in hospitals due to the burden of inventory management and logistics.<sup>16</sup> The burdens could cause clinicians to lose motivation when additional work is unrecognized and is not billable.<sup>17</sup> With the increased smartphone penetration,<sup>18,19</sup> many researchers turned to the "Bring Your Own Device" (BYOD) strategy.<sup>20</sup> Pugliese et al.<sup>21</sup> found that BYOD demonstrated a higher engagement compared to the provisioned device. Although BYOD brings more freedom and choices to end users,<sup>22</sup> security concerns arise in the bidirectional flow of information between mobile apps and EHRs.<sup>23</sup> Most EHRs are reluctant to expose application programming interfaces (APIs) for writing permissions.<sup>24</sup> To receive PROs from BYOD, Genes et al.<sup>25</sup> developed a mobile collection app, which relied on a secure agent (i.e. portal client) to communicate with EHRs. The approach works well for an individual

provider; however, it might be burdensome when patients interact with multiple different EHRs. It is also possible to have BYOD directly connected via wireless in hospitals, but the security and logistics concerns are as high as patient portals.<sup>26,27</sup>

During COVID-19, quick response (QR) codes are widely used for in-office surveys.<sup>28–30</sup> Technically, the symbol encodes a Uniform Resource Locator (URL) to a remote survey page. Directing a smartphone's camera to it would quickly open the survey page in the default web browser if it is not password protected. The camera works like a QR code reader, which receives the symbol content across the air. Unlike networks, the QR-mediated data transmitting often involves a human. The setup does not rely on any tedious steps, such as API endpoints and authentication. Moreover, it is social distancing friendly, as a survey request can be dispatched without any physical contact. It is a ubiquitous modality to be used to channel data onto mobile devices, however, using it to send data to EHRs, especially for PROs is seldom studied.

### Study focus

The current methods of PRO collection are either technology intensive or labor intensive. To sustain engagement, it is hard to attend to one thing and lose sight of another.<sup>31</sup> The objective of this study is to use QR technologies to reduce PRO integration burdens for both patients and clinicians. A modular and shared FFP was proposed in this study, and FFC was implemented as an EHR-agnostic component. Then, the integration was proposed for PROs over a QR code (PROoQR). After coding and testing, the framework was evaluated in a real clinical setting.

## Methods

### Framework overview

In China, PRO integration just started a few years ago. A broad solution was to collect PROs independently outside of the EHR in a separate system. Both hospitals and patients had security concerns when PROs were circulated online. It was a part of the reason why the online completion rate remained lower. Then, tablets were used as a major modality for PRO collection. However, it placed tremendous burdens on clinicians.

It happened too in the hospital of the study team. Complaints were there for a while, especially during COVID-19. It was urgent to find a better solution to in-office assessments. First, BYOD devices were tried, as almost every patient carried a smartphone for a visit. It didn't work well due to some barriers such as double log-ins and bad signals. Then, an interactive voice response system was deployed in the waiting room.<sup>32</sup> However, it failed again because of the

noisy environment and extended completion time. Finally, QR technologies came into view.

As above mentioned, mobile devices equipped with a QR code reader were able to receive data from a static symbol of QR codes. The directional flow was reversed in this study, namely, the data flow from a mobile phone to a stationary desktop computer. The encoded content in QR codes was often static, for example, URL. Conversely, this study put dynamic content (i.e. PRO results) into a QR code. An overall workflow was proposed in Figure 1: (1) to deploy a new PROM, a clinician uploaded a well-designed web page onto a cloud repository such as GitHub, (2) Prior to the visit, a PRO request (i.e. URL) was dispatched to a patient via reminding messages, (3) the patient started to fill the assessment form on his or her smartphone, (4) when submitted, a QR code was generated which contained the submitted result, (5) the patient walked in and had the QR code scanned in the encounter room, (6) the data was received and automatically populated into the discrete fields of FFC, (7) the clinician started to review PROs and discussed them with the patient face-to-face, (8) the clinician annotated PROs or worked with the patient to modify if needed, and (9) the clinician clicked a button to accept the data and to persist the captured data into the EHR.

The framework consisted of two modular systems. Both of them were open-sourced on GitHub. One was for FFP (<https://github.com/PROoQR/prom>), which was a public site that patients were able to access from anywhere. The other was for clinicians (<https://github.com/PROoQR/ehr-pro>), which was deployed in an intranet setting. As illustrated in Figure 2, both of them were built on a model-view-controller pattern. At the model layer, JavaScript Object Notation (JSON) was used to describe a

common data model. At the view layer, Hyper Text Markup Language (HTML) was used for both of them. Each PROM used a different HTML file. At the controller layer, both of them were built on JavaScript. For FFP, the controller was implemented as a library. It can be directly included in the HTML files. With the shared controller, a PROM customization can be easily done by modifying the HTML file only. Each HTML file of FFP worked separately like a single-page application (SPA). A change made in one file wouldn't impact another. Being a SPA, it offered faster transitions that made the normal web page more like a native app. FFP was not driven by any database. The only output file was a QR code, which was used as an input for FFC. FFC was set up in the hospital's network and used a database to persist the PRO data sent from FFP. A change made to the model of FFP had to be synchronized to multiple FFCs.

### Data model

A shared data model was basically a JSON file, which consisted of attribute-value pairs and array data types. It was easy to read so that it can be modified even by a person who had no programming background. It was used to describe some data elements for a survey. At FFP, the JSON models were directly included in the HTML files. At FFC, JSON models were mapped to some object models (Figure 3), which were coded using JavaScript. Both of the models followed a hierarchical structure. Namely, a survey object had a couple of sections, a section included a group of questions, and a question had several answers. Each answer was an enumerable value and was mapped to a relevant Likert scale.

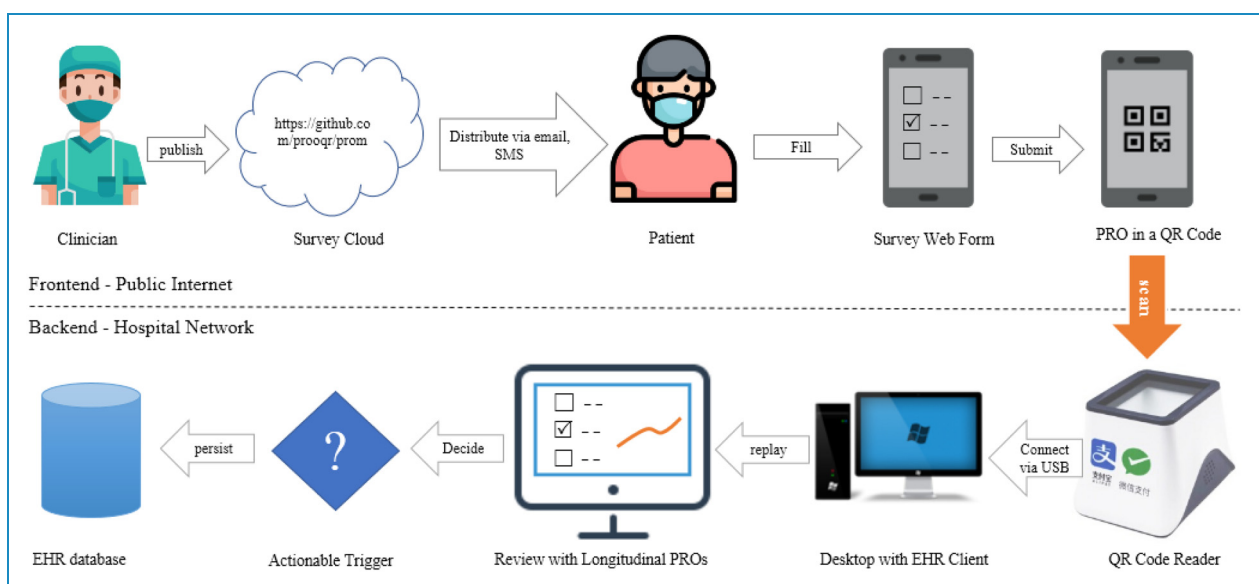


Figure 1. Framework overview.

Given the volume of survey elements, there were two ways to render it.<sup>33</sup> One was to use a single-page testing (SPT) by rendering all items in a short form. The other

was a computerized adaptive testing (CAT), which used item response theory to render the next item based on the most recent answered items. Compared to SPT, CAT significantly saved time when a survey was too long. As shown in Table 1, both of them were supported via a “type” attribute in the models. For CAT-based surveys, some attributes were provided to describe how elements were programmatically rendered (Figure 4). For example, a “goto” attribute was assigned to the answer object, and another attribute or “on” was assigned to the section object. Selecting an answer could trigger an event specified by “goto.” When an event was triggered, some relevant sections should be available for display. The framework went through every “on” attribute to find the relevant sections for the named event. Then it displayed them to patients in order. The “on” attribute was an array because a section could listen to multiple events. In addition, a “score” attribute was assigned to an answer object. In accordance with practice, the score must be a numeric value. When an answer was selected, the score was calculated at as the question level. A score of a section was a summed value

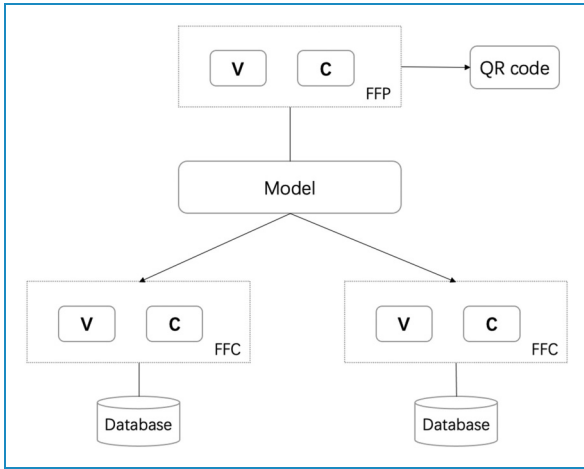


Figure 2. The architecture of the framework.

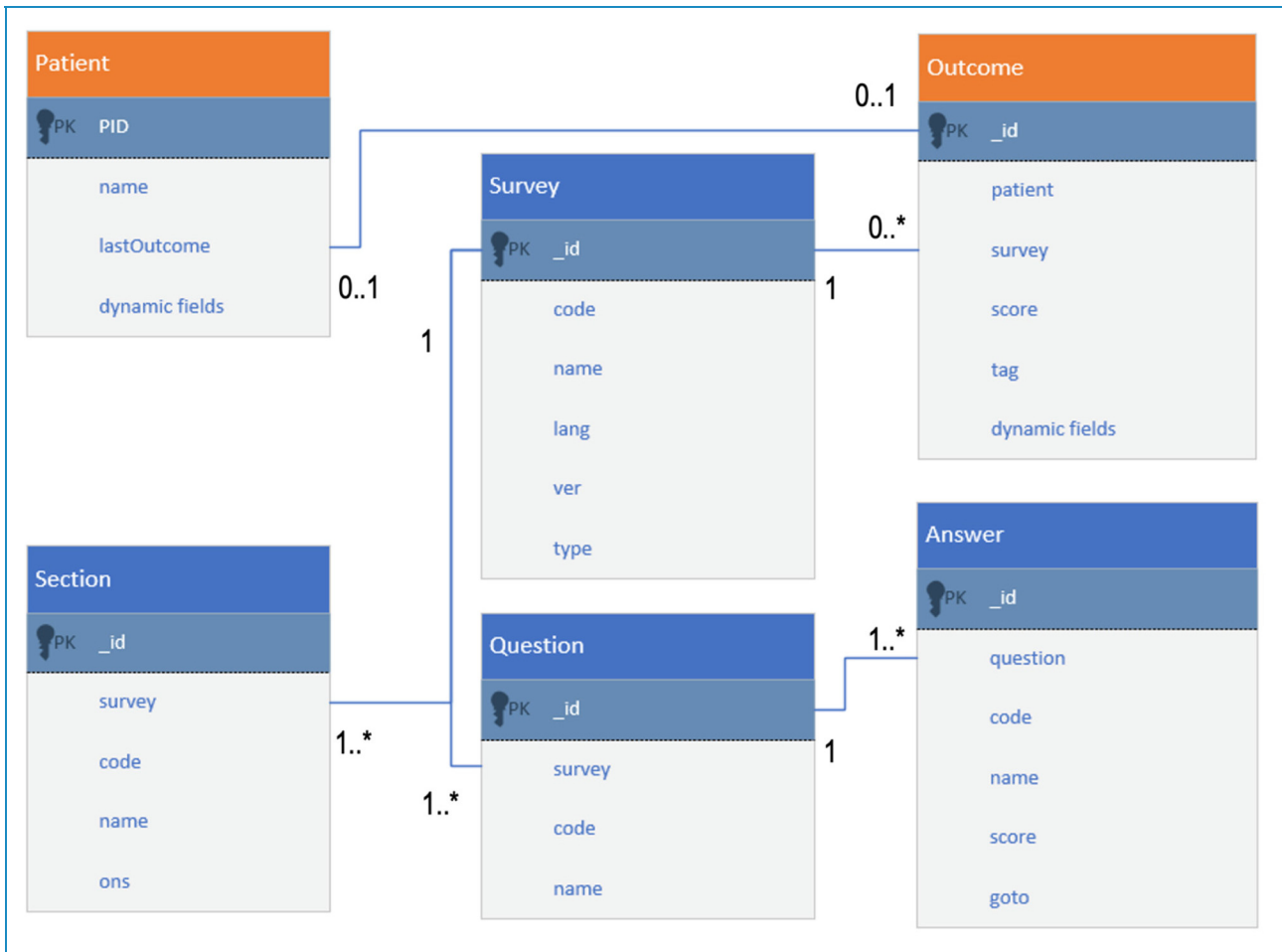


Figure 3. Data model.

**Table 1.** JSON elements for a PROM.

Object	Attribute	Description	Example
<b>Survey</b>			
	code	Survey code	WOMAC
	name	Survey name	The Western Ontario and McMaster Universities Arthritis Index (WOMAC) <sup>34</sup>
	lang	Language code	en/zh-CN
	ver	Version number	1
	type	Survey rendering type	SPT/CAT
	sections	Array of section objects.	
	questions	Array of question objects	
<b>Section</b>			
	code	Section code	A0
	name	Section name	Pain
	on	List of trigger names	hip, neck
<b>Question</b>			
	code	Question code	A0A
	name	Question name	Walking on flat surface?
	answers	Array of answer elements	
<b>Answer</b>			
	code	Answer code	A
	name	Answer name	Moderate
	score	Numeric value	2
	goto	An event or trigger name	hip

JSON: JavaScript Object Notation; PROM: patient-reported outcome measure; SPT: single-page testing; CAT: computerized adaptive testing.

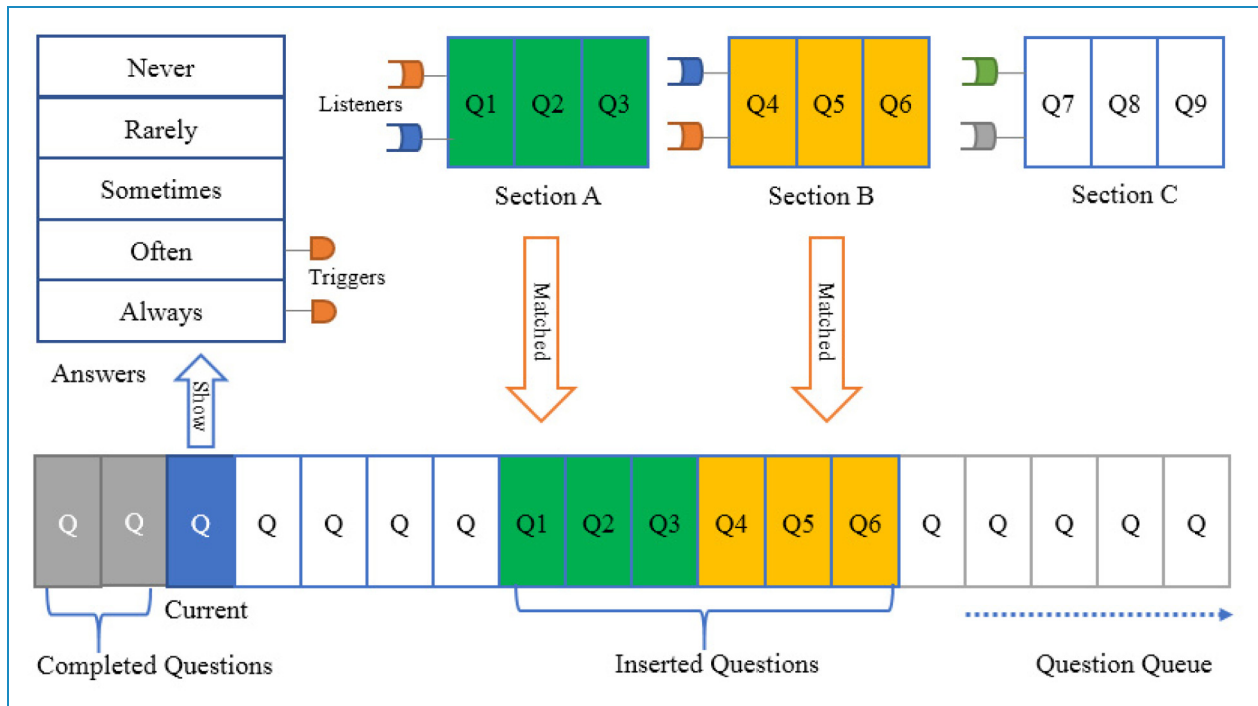
for all of the subordinate questions. Similarly, a score of a survey was a total after calculating the section scores. The scores were calculated at the runtime and saved into database. They were not included in the QR codes during data transmitting.

### QR code data structure

The capacity of a QR code depends on the error correction levels (e.g. low, medium, quartile, and high).<sup>35,36</sup> When smartphones' screen had some dirt or cracks, it might

block the scanning. At the medium level, when the damage was less than 15% of the symbol, the code was still readable. In this study, the medium level was chosen to have an error resistance. QR codes can also encode different types of characters (e.g. numeric, alphanumeric, kanji, and byte). Alphanumeric characters were used in this study to encode structured PRO data. Under the assumption, the maximum length of the encoded characters was 3391.<sup>35</sup>

To fit in the limited capacity, the framework used some alphanumeric codes for QR storage rather than literal



**Figure 4.** Computerized adaptive testing.

**Table 2.** The rules to code a patient-reported outcome measure (PROM).

Object	Length	Description	Combinations
Survey	Up to 10	To ensure a PROM unique globally.	unlimited
Section	2	To code a unique section given a PROM. The length was fixed.	1296 (36*36, using only 0-9 and A-Z) per survey
Question	3	The length was fixed. The first two characters were the code of the parent section.	36 per section 46,656 per survey
Answer	1	To code an enumerable answer given a question.	36 per question

values. So, the short codes consisted merely of 0–9 and A–Z.<sup>35,36</sup> Since a real-world questionnaire has limited items, a QR-based data structure was proposed based on some rules (Table 2). To ensure consistency, the naming should be handled properly. For example, when “moderate” was

used to label an answer, the code could be reserved for “A,” even if it was in a different questionnaire.

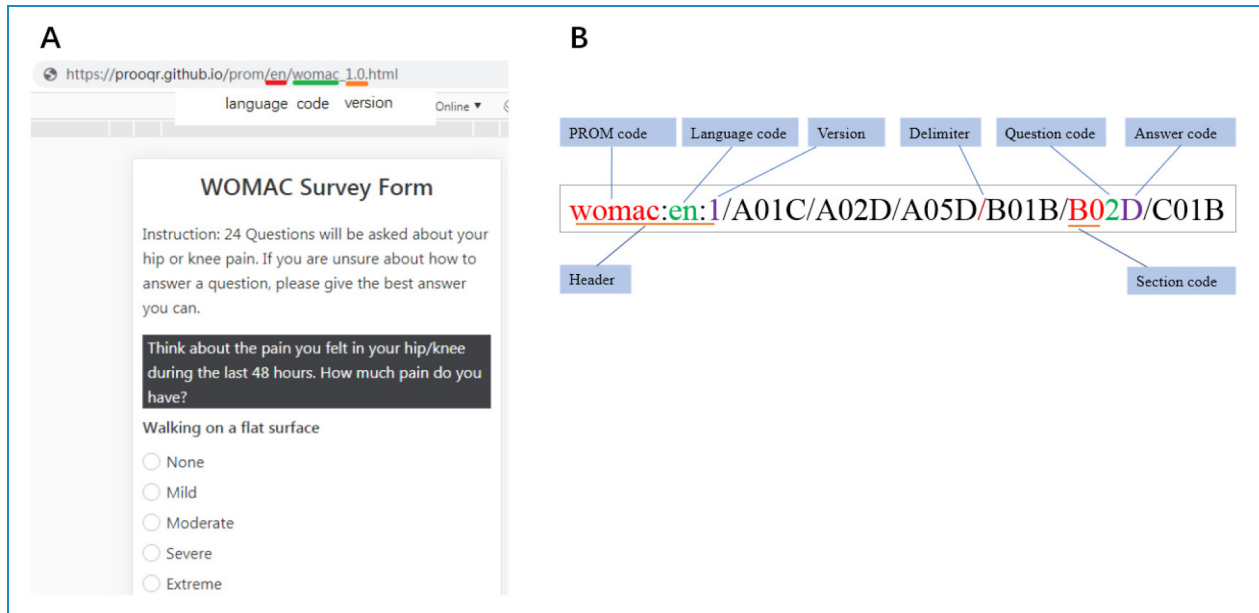
Apart from the “code” attribute, every survey object was assigned two more version-related attributes (“lang” and “ver”) as shown in Table 1. All of the three attributes were grouped together as a header, as illustrated in Figure 5. Thus, different PROMs owned a different header. Unlike JSON or other standard formats, the framework concatenated each of the selected elements based on the short codes into a compacted text (Figure 5).

### QR code reader

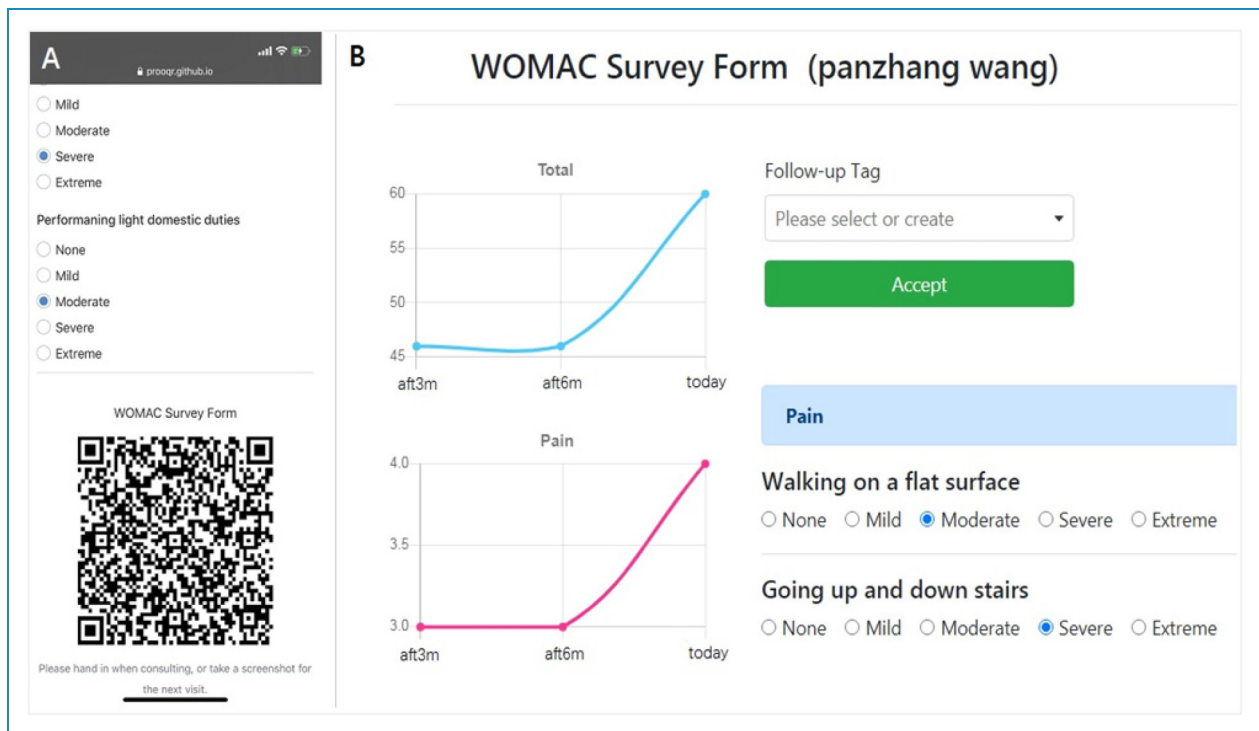
At the point of care, a QR code reader was attached to a desktop computer where FFC was installed (Figure 1). It was plug-and-play and easy to install. It was placed somewhere on the desk in the encounter room where patients can easily reach when they walked in. With the survey result saved in a QR (Figure 6), patients were asked to sway it over the QR code reader when approaching the desk. The reader worked exactly like a smartphone’s camera; however, clinicians don’t have to operate it proactively. So, clinicians could talk with patients while looking at the readings on the screen faced to them.

### Questionnaire synchronization

As per the guide, when patients were seen in more than one clinic, questionnaires should be consistent across areas. In



**Figure 5.** Data structure. (A) URL structure. (B) The encoded data format for a QR code. URL: uniform resource locator; QR: quick response.



**Figure 6.** Screenshots (A) convert a survey into a QR code (B) PRO review. QR: quick response; PRO: patient-reported outcome.

this study, any change to the JSON model had to be made on the shared FFP first FFC would periodically sync with FFP by downloading an updated version. The synchronization could be handled offline as well when FFC was

disconnected from the internet, by manually downloading. Since most of the questionnaires had less changes after being finalized, it wouldn't be too often for the manual synchronization. To exchange collected data between

providers, some backend synchronizations could be established for help.<sup>10</sup> However, since patients owned the original data, it was feasible to read from their smartphones directly.

### *Reminder and notification*

Clinicians could prescribe a remote questionnaire for a scheduled appointment, as per the guide, the prescription was sent to patients as a reminder prior to the visit. Email and text are two popular modalities used on smartphones to deliver reminders. When a reminder reached patients using the predetermined modality, a link was presented to patients. Clicking on it would bring patients to the shared FFP for assessments. When completed, the result was saved into a QR code. However, according to the guide, when a scoring threshold was crossed, providers should be notified in time. To do that, a URL was appended as a query parameter to the enclosed link. The URL was a server address, which was set up to receive notifications. Apart from the server's name, a nonce token was appended to the URL. Being a universal unique code, it was generated randomly and can be used only one time within a short life time. It had as long as 21 alphabet symbols, but remained URL friendly. When the server received a response, it first tried to verify the token. If it was valid, a six-digit number was sent back via text. Patients could easily input the received number on their mobile phones. Providers would be notified using the raw content (Figure 5), only when the number was matched.

The alerts were transparent to patients. FFP would let patients know when an alert was ready to send. So, patients would understand their choices and some errors could be avoided before that. After the button was clicked, a text box appeared on the screen to allow patients to enter the identifying code. The alerts were open-ended in FFP. Providers would be notified, but electronic acknowledgments were not warranted. On the other hand, patients were able to mute the notification by bringing the code to clinicians in person.

For questionnaires sent from some clinical registry systems, it did not involve a face-to-face consultation. Remote collections were necessary. In this case, the framework used the notification service to facilitate the collection. An additional parameter was added to the URL, which informed FFP that notifications would be triggered unconditionally. Once the registry server received the results, an acknowledgment could be addressed to patients using the predetermined modality.

Questionnaires might not always be administered remotely. Sometimes, patients walked in without a QR code. To provide "just-in-time" support, a receptionist could help patients to quickly generate a QR code at check-in. Instead of providing a tablet or asking for an email, the receptionist dispatched a PRO request directly

onto the screen that faced to patients. It was also shown as a QR code. Hence, patients could start an assessment right away by scanning it.

### *In-office PRO review*

After patients submitted the data on the QR code reader, a survey form would automatically display on the screen for clinicians. The form fields were auto-populated with the captured data. Clinicians were able to annotate the data (e.g. tags) before accepting it into the EHR (Figure 6). Since patients owned the data, if a change was necessary, it had to start from patients. Then the modification required a rescan. To save the encoded content into database, it was important to make sure it was associated with the right patient identifier (PID). The encoded data was completely anonymized. To preserve private or confidential information, the framework removed any identifiers from the code that could link individuals to the stored data. When the "Accept" button was clicked (Figure 6), the anonymized data was saved and associated with the PID under the clinicians' eyes. The data was assigned to a record denoted by the PID no matter who scanned it.

### *PRO data display*

Given PIDs, most of the captured data was stored with temporal values in the FFC database only. A semantic tag created during review could be used for temporal values (Figure 6). Some demographic information was fetched from EHRs and saved redundantly in FFC. Based on the aggregated data, FFC used Chart.js to visualize PROs. For example, during the PRO review, longitudinal data was displaying at the individual level to help clinicians with follow-up care.<sup>37</sup> At the population level, FFC allowed clinicians to run some customized queries to show longitudinal trends to facilitate comparative analysis.

### *Module development*

FFP was developed using the Svelte framework,<sup>38</sup> which is a JavaScript framework to build faster SPAs with a significantly reduced bundled file. A controller was written in modern JavaScript. Then, it was bundled into a compact file to be included for HTMLs. HTMLs consisted of a snippet where a global variable was defined to point to the shared JSON model. The global variable can be accessed from the bundled controller. For maintenance purposes, the snippet was included in plain text only rather than compiled. Each version of a PROM was deployed separately; however, they shared the same logic or the controller file. PROMs were able to deploy on any website such as the GitHub Pages, which is a popular static website for open-source projects.



FFC was a web-based application built on the Express.js framework.<sup>39</sup> One feature of the framework was to allow clinicians to create new PROMs with user-defined codes (Table 2) on the fly. To do that, NoSQL<sup>40</sup> was used to persist the schema-less PRO data. As a NoSQL database, MongoDB was chosen, because it can work just like a plug-in database to a relational database.<sup>41</sup> Mongoose was used to model the MongoDB objects (Figure 3) in the controller based on Node.js.<sup>41</sup>

At the backend, according to the guide, PID was used to integrate FFC with the EHR. In the data model, PID was also used as a foreign key (Figure 3). To a legacy EHR client, FFC pages were pluggable. Given a PID, the integration could be triggered on some events. For example, to display longitudinal data, a button or a menu had to be added to the EHR client. Once clicked, PID was sent as a query parameter to a URL of FFC, and then a web page showed up with some history data.

### Evaluation method

The study team's hospital had three campuses, which all preferred tablets for PRO collection. For study purposes, evaluation environments were set up from the ground up as shown in Table 3. The Western Ontario and McMaster Universities Arthritis Index (WOMAC) questionnaire<sup>34</sup> was chosen as the studied instrument. Five clinicians were recruited for each campus, who had no former experience with PRO collection. After that, they were trained to work with on-duty physicians for routine integration; however, they were limited to the named approach on each campus. For eligible patients, they were randomly selected to participate in the named experiment on each campus. After the assessment was done, each participant was asked for two optional questionnaires. One was the NASA Task Load Index (NASA-TLX)<sup>42</sup> used to measure

the workload, the other was the Net Promoter Score (NPS)<sup>43</sup> used to measure the user satisfaction. The question for NPS was like "On a scale from 0–10, how likely are you to use the method to contribute your PRO data in the future?" On the basis of response, participants were classified as potential system promoters (response score: 9–10), passives (response score: 7–8), or detractors (response score: 0–6).  $P_p$  was used to denote the percentage of promoters.  $P_d$  was used to denote the percentage of detractors. The NPS was denoted by  $S = P_p - P_d$ . System logs were used to calculate the total time for an assessment. For example, the completion time for tablets included time spent by a patient and a clinician. The response rate was used to monitor how many assessments were completed followed by a reminder. Data was also collected for patients who had completed a remote assessment but ended with a no-show. For study purposes, the data generated remotely in the framework group was collected via the notification service.

After the setup, some experts were invited for verification and validation. The evaluation started after that and it lasted 3 months. Then, the participated clinicians were asked to complete NASA-TLX and NPS surveys. The NPS question was like "On a scale from 0–10, how likely are you to use the method to collect PRO data in the future or recommend it to others?"

## Results

### Storage capacity

Based on the coding system, as shown in Table 2, there was almost no limitation to code a PROM. Given a PROM, the model can support up to 46,656 questions. Each question supported up to 36 answers. In the real world, SPT-based PROMs tend to have a very limited length. For example, there are only 42 items for a Knee injury and Osteoarthritis Outcome Score<sup>44</sup> and only 24 items for WOMAC. Even a CAT-based questionnaire<sup>33</sup> has around 100 questions. Typically, each question contains an odd number of answers no more than 7. Thus, the JSON model and coding system were sufficient to meet the real-world need. According to the rules (Table 2) and the QR code data structure, the capacity for a QR code was measured. The results are shown in Table 4. It showed that a QR code could contain up to 1120 items. The capacity of QR-based storage was also sufficient to meet the real-world need.

### Facilitators

As aforementioned, JSON and GitHub solidified the framework. They were combined to shed their inherent benefits on the solution. Historically, databases are used for PROMs customization.<sup>45,46</sup> The database-driven solution lacks interoperability because it tends to bind the

**Table 3.** Clinical testing environments.

Campus	Modality	Minimal environment setup
A	Tablet	A simple web application was developed to be used internally for tablets-based integration.
B	Portal	A cloud server was purchased. Based on it, a simulated portal was developed, which included functions like survey, registration, login and some menus for navigation. A simple FFC was developed as well.
C	PROoQR	Some localization was made to the proposed framework.

FFC: frontend for clinicians; PROoQR: PRO over QR code.

**Table 4.** Capacity of quick response (QR)-encoded items.

Item	Length	Description
QR code storage limit	3391 characters	As assumed above
Header	Up to 30 characters reserved	Non-fixed (Figure 5)
Surveyed item	Each was fixed as long as 5 characters	2 characters for a section, 1 for a question, 1 for an answer, and 1 for a delimiter (Figure 5)
Capacity of items	1120	(3391-30)/5

customization with a specific EHR platform. JSON fits better in terms of interoperability due to its readability and portability.<sup>47</sup> GitHub is a famous and robust website for hosting open-source projects and static websites. GitHub also facilitates collaboration via a workflow to manage pull requests (PRs).<sup>20</sup> With these, this study proposed a standardized procedure for PROMs customization: (1) creating a PR to duplicate an existing PROM, (2) changing the file name and JSON content accordingly, (3) reviewing the PR, and (4) accepting the PR for auto-deployment. The procedure enabled non-developers to contribute PROMs and let the community to review or reuse. Since the front end was GitHub-based and EHR-agnostic, a shared web-based PRO collection system and consistent interfaces were easily achieved. As a direct result, it built a common ground toward the PRO-based interoperability.

### Evaluation outcomes

Compared to the framework, as shown in Table 5, the tablet group took 2.3 times of effort for implementation, whereas the portal group took 8.3 times of effort. To purchase a QR code reader, it is only \$50. By comparison, a tablet cost about \$600, and a cloud server cost \$2000.

The total number of responses collected from patients was also listed in Table 5. Based on them, the assessment results were shown in Table 6. When comparing the response rate for remote participants, the result showed that the framework group was higher than the portal group (76.6% vs. 61.1%). For completion time, the framework group saved 37.5% of time when compared to the tablet group and saved 43.4% of time when compared to the portal group. The average workload score was calculated as well. When using the framework, patients reported a lower workload (23.2 vs. 47.8) when compared to the

portal group ( $p < .001$ ), and clinicians reported a lower workload (16.3 vs. 44.9) when compared to the tablet group ( $p < .001$ ). Among the studied modalities, the NPS for the framework group was the highest for both clinicians (89.3%) and patients (86.5%). The no-show rate was reduced (8.6% vs. 11.7%) when compared to the portal group.

## Discussion

### Principal results

When integrating PROs in EHRs, both the portal- and tablet-based approaches are facing challenges. The study team lifted some barriers for patients and clinicians by advancing QR codes in the process of electronic PRO integration.

A successful PRO integration often starts with a viable process of data collection. Previously, it is either self-administered (i.e. patient portals) to patients or assisted (i.e. tablets) by clinicians. They are separately burdened with logistic issues in the routine process. The double squeeze might dampen the sustained engagement and lead to a slower adoption than expected.<sup>48,49</sup> The study suggested a modular approach for PRO integration. Both FFP and FFC were designed as lean as possible, so patients and clinicians could get rid of the tedious part and focusing on their core business. QR code technologies were proved to be good enough to transmit data between them. Compared to the prevailing methods, the framework had several improvements, such as implementation effort, completion time, response rate, workload, and user satisfaction. Surprisingly, the no-show rate was reduced when using the framework remotely.

### Comparison with prior works

The prior solutions are often resource intensive. Patient portals are piled up with not only hundreds of features but also cutting-edge technologies.<sup>50,51</sup> Using tablets is labor-intensive because of the inventory management and logistics. For a new site deployment, both of them involve a lot of effort and money. On the contrary, the framework is faster and cheaper. Firstly, it was effortless and cost less to set up an FFP, as it is a shared website with static pages. Secondly, it takes less effort to deploy an FFC, since it is EHR-agnostic and can be easily bolted on to an existing system. Lastly, a QR code reader is as cheap as \$50.

It is necessary for a modality to fit in the clinical workflow to pursue a sustained integration.<sup>31,52</sup> Filling out a form and reviewing results are two important steps. Other than abnormal data, both clinicians and patients favor a face-to-face review<sup>31</sup> in routine care. When the first step is performed remotely, it is asynchronous to the last step

**Table 5.** Implementation effort and responses from patients.

Modality group	Effort in man-days	Reminders dispatched	Total number of PRO assessments	No-shows	Total number of workload assessments	Total number of NPS assessments
Tablet	7	0	345	0	206	225
Portal	25	365	223	26	167	151
PROoQR	3	351	Remote: 186 In-office: 83	23	211	233

PRO: patient-reported outcome; PROoQR: PRO over QR code.

**Table 6.** Evaluation outcomes.

Modality group	Average workload score for patients	NPS for patients	Average workload score for clinicians	NPS for clinicians	Average minutes for completion	No-show rate
Tablet	20.5	82.3%	44.9	54.2%	Clinicians: 2.0 Patients: 2.8	
Portal	47.8	60.9%	15.5	77.7%	5.3	11.7%
PROoQR	23.2	86.5%	16.3	89.3%	Clinicians: 0.2 Patients: 2.9 Total: 3.0	8.6%

NPS: Net Promoter Score; PROoQR: PRO over QR code.

as it causes less disruption. Since patient portals are web-based, during the first step, all of the PROs are submitted directly to the remote EHRs. When a booked visit ends with a no-show, patients might not get a chance to touch the uploaded PROs. Rearrangements often come with new PRO requests. Thus, some PROs might be left unreviewed forever. To have those PROs accepted as much as possible, it needs a frequent “check-the-box” review. Doing it will cause “alert-fatigue” or “information overload” to clinicians.<sup>53</sup> The framework handles submission in a different way. Abnormal values alert clinicians in real time, and normal PROs were paused for submission. The paused PROs will not get into the box until they are brought in for a face-to-face review. It helps to reduce the overload and mitigate the risk that no-shows put on the quality of the overall dataset. There are many reasons for a no-show, but the reduced no-show rate (11.7% vs. 8.6%) suggests that some patients tend to complete the last step. The first step can be conducted synchronously at the visit as well. The synchronous collection ensures that data collected at check-in would be reviewed as much as possible. Doing it on tablets slightly reduce patients’ workload scores (20.5 vs. 23.3) when compared to the framework group. However, it requires additional devices and needs clinicians’ effort to physically operate the devices. Doing it via QR codes avoids physical contact and additional devices. A PRO request could start with a QR code and end with another QR code during check-in.

Clinicians barely move or have to remember where the devices are. The completion time for clinicians is significantly reduced (0.2 min vs. 2.0 min) when compared to the tablet group. Their workload is reduced ( $p < 0.001$ ) as a result. To sum up, the current modalities are biased, being either synchronous or asynchronous. The framework gains an advantage from both sides by presenting a mixed mode with two visible steps. The high NPS (clinicians: 89.3%, patients: 86.5%) scores suggest that every stakeholder is happy with it.

Patients often have concerns about security and privacy when logging into portals or transmitting data over networks.<sup>54,55</sup> In recent years, U.S. healthcare data breaches are increasing at alarming rates.<sup>56</sup> The situation is improved after employing some technologies, such as two-factor authentication<sup>57,58</sup> and CAPTCHA.<sup>59</sup> However, it might annoy users and make the collection burdensome. PRO integration cannot work as expected if portals turn off authentication and authorization. However, the framework suggests a way to function when the log-in is not available. When PROs are collected on a shared static website, some additional pages become unnecessary, such as registration and password retrieving. Because of the simplified log-ins, the completion time for patients is saved by 45.3% and patients’ workload is also reduced ( $p < 0.001$ ) when compared to the portal group. With the easy-to-access pages, the response rate is improved (76.6% vs. 61.1%). Apart from that, normal PROs are transmitted off the

grid. It helps protecting patient privacy as the data is brought in person to clinicians without any intermediary.<sup>60</sup> The data can be safely and correctly attached to the right record in EHRs under clinicians' eyes.

In some cases, providers might want to replace an existing PRO system with a new one, since there are so many vendors out there to choose. It seems a small change, but it never happens smoothly in the health information economy.<sup>61</sup> Most PRO systems are highly integrated. Substitutability is only warranted if vendors provide modular services with common data elements.<sup>24,61</sup> An promising move is to invest in Fast Health Interoperability Resources (FHIR),<sup>62</sup> which aims to standardize a group of resources and APIs. Unlike FHIR, the framework discourages APIs and promotes modular services. Instead of using long and semantic properties, the framework uses short and non-standardized codes to fit in the limited capacity of QR codes. Abandoning APIs helps reducing complexity for integration and needs little effort to implement. When PROs are transmitted between systems via QR code readers, the interoperability is assured on a channel that is driven by humans rather than APIs. The flexibility might advance substitutable apps on mobile devices.<sup>63</sup> For example, Personal Health Records (PHRs)<sup>64</sup> can be enhanced by incorporating the static FFP. Instead of interacting with multiple EHRs, patients can choose a single PHR to manage their PROs while being loosely connected. Thus, as consumers, patients, and clinicians could remain engaged while mobile apps compete on value and cost. It looks disruptive and needs a culture change; however, the team believes the paradigm shift will be human friendly, especially at the point of care. It won't take much time and effort to normalize.

## Limitations

This study has several limitations. First, QR code is designed to store only structured PRO data, so storing long unstructured data would easily deplete the capacity. Second, a survey question is designed to have a single answer to help score calculation. For multiple choices, a special delimiter could be placed between encoded items. But score calculation would be complicated, and a simple sum might not be the right answer. Third, due to the limited time, PROM customization and interoperability are not evaluated in this study. Fourth, the framework is limited to a mobile web application. Future works include integrating it into mobile PHRs to manage PRO integration. Last, it is not a full-blown product and is mainly implemented as a reference to elucidate the QR-mediated integration. More enhancements are warranted in the future.

## Conclusions

Applying QR codes and QR code readers in the process of PROs integration can help mitigating burdens for patients

and clinicians. The framework is workflow-friendly and helpful for protecting patient privacy and data security. QR-mediated data exchange at the point of care presents an alternative to pursue substitutability and interoperability.

**Conflicting of interest:** The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Contributorship:** PW, TL and LY designed the study. PW developed the model and prepared figures. TL and LZ performed the evaluation and assembled the data. PW, TY, and LY wrote the manuscript. All authors contributed to the final approval of the manuscript.

**Ethical approval:** Not applicable.

**Funding:** The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported Shanghai Jiao Tong University Affiliated Sixth People's Hospital (grant no. DY2019015).

**Guarantor:** Not applicable.

**ORCID iD:** Panzhang Wang  <https://orcid.org/0000-0002-2743-4904>

## References

1. Basch E. Patient-reported outcomes – harnessing patients' voices to improve clinical care. *N Engl J Med* 2017; 376: 105–108.
2. Leahy AB and Steineck A. Patient-reported outcomes in pediatric oncology: the patient voice as a gold standard. *JAMA Pediatr* 2020; 174(11): e202868.
3. Leblanc TW and Abernethy AP. Patient-reported outcomes in cancer care – hearing the patient voice at greater volume. *Nat Rev Clin Oncol* 2017; 14: 763–772.
4. Woods SS, Evans NC and Frisbee KL. Integrating patient voices into health information for self-care and patient-clinician partnerships: veterans affairs design recommendations for patient-generated data applications. *J Am Med Inform Assoc: JAMIA* 2016; 23: 491–495.
5. Zhang R, et al. Provider perspectives on the integration of patient-reported outcomes in an electronic health record. *JAMIA Open* 2019; 2: 73–80.
6. Gensheimer SG, et al. Oh, the places we'll go: patient-reported outcomes and electronic health records. *The Patient – Patient-Centered Outcomes Res* 2018; 11: 591–598.
7. Lewinski AA, et al. Bridging the integration gap between patient-generated blood glucose data and electronic health records. *J Am Med Inform Assoc: JAMIA* 2019; 26: 667–672.
8. Research, U.S.D.o.H., et al. Guidance for industry: patient-reported outcome measures: use in medical product development to support labeling claims: draft guidance. *Health Qual Life Outcomes* 2006; 4: 79–79.

9. Weldring T and Smith SMS. Patient-reported outcomes (PROs) and patient-reported outcome measures (PROMs). *Health Serv Insights* 2013; 6: 61–68.
10. Ali J, et al. Users' Guide to Integrating Patient-Reported Outcomes in Electronic Health Records, 2017.
11. Tieu L, et al. Online patient websites for electronic health record access among vulnerable populations: portals to nowhere? *J Am Med Inform Assoc* 2017; 24: e47–e54.
12. Tieu L, et al. Barriers and facilitators to online portal use among patients and caregivers in a safety net health care system: a qualitative study. *J Med Internet Res* 2015; 17(12): e275.
13. Lyles CR, et al. Using electronic health record portals to improve patient engagement: research priorities and best practices. *Ann Intern Med* 2020; 172: S123–S129.
14. Bauer AM, et al. Aligning health information technologies with effective service delivery models to improve chronic disease care. *Prev Med* 2014; 66: 167–172.
15. Portz JD, et al. Using the technology acceptance model to explore user experience, intent to use, and use behavior of a patient portal among older adults with multiple chronic conditions: descriptive qualitative study. *J Med Internet Res* 2019; 21(17): e11604.
16. Wong D, et al. Exploring the use of tablet computer-based electronic data capture system to assess patient-reported measures among patients with chronic kidney disease: a pilot study. *BMC Nephrol* 2017; 18(1): 356.
17. Lv N, et al. Personalized hypertension management using patient-generated health data integrated with electronic health records (EMPOWER-H): six-month pre-post study. *J Med Internet Res* 2017; 19(19): e311.
18. Austin L, et al. Providing 'the bigger picture': benefits and feasibility of integrating remote monitoring from smartphones into the electronic health record. *Rheumatology (Oxford, England)* 2019; 59: 367–378.
19. Chan Y-FY, et al. The asthma mobile health study, a large-scale clinical observational study using ResearchKit. *Nat Biotechnol* 2017; 35: 354–362.
20. Gwaltney C, et al. "Bring your own device" (BYOD). *Ther Innov Regul Sci* 2015; 49: 783–791.
21. Pugliese L, et al. Feasibility of the "Bring Your Own Device" model in clinical research: results from a randomized controlled pilot study of a mobile patient engagement tool. *Cureus* 2016; 8(3): e535.
22. Portela F, Veiga AMd and Santos MF. *Benefits of Bring Your Own Device in Healthcare*. 2018.
23. Lomotey RK and Deters R. *Mobile-Based Medical Data Accessibility in mHealth*. 2014 2nd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, 2014; p. 91–100.
24. Gordon WJ and Mandl KD. The 21st century cures act: a competitive apps market and the risk of innovation blocking. *J Med Internet Res* 2020; 22: e24824.
25. Genes N, et al. From smartphone to EHR: a case report on integrating patient-generated health data. *NPJ Digit Med* 2018; 1(01): 23.
26. Ahmed SM. *BYOD, Personal Area Networks (PANs) and IOT: Threats to Patients Privacy*. in *RIIFORUM*. 2019.
27. Wani TA, Mendoza A and Gray KM. Hospital bring-your-own-device security challenges and solutions: systematic review of gray literature. *JMIR Mhealth Uhealth* 2020; 8: e18175.
28. Kernder A, et al. Digital rheumatology in the era of COVID-19: results of a national patient and physician survey. *RMD Open* 2021; 7: e001548.
29. Pérez-Alba E, et al. Use of self-administered surveys through QR code and same center telemedicine in a walk-in clinic in the era of COVID-19. *J Am Med Inform Assoc: JAMIA* 2020; 27: 985–986.
30. Whitelaw S, et al. Applications of digital technology in COVID-19 pandemic planning and response. *Lancet. Digit Health* 2020; 2: e435–e440.
31. Reading MJ and Merrill J. Converging and diverging needs between patients and providers who are collecting and using patient-generated health data: an integrative review. *J Am Med Inform Assoc: JAMIA* 2018; 25: 759–771.
32. Heisler M and Piette JD. I help you, and you help me. *Diabetes Educ* 2005; 31: 869–879.
33. Hung M, et al. Computerized adaptive testing using the PROMIS physical function item bank reduces test burden with less ceiling effects compared with the short musculoskeletal function assessment in orthopaedic trauma patients. *J Orthop Trauma* 2014; 28: 439–443.
34. Bellamy N, et al. Validation study of WOMAC: a health status instrument for measuring clinically important patient relevant outcomes to antirheumatic drug therapy in patients with osteoarthritis of the hip or knee. *J Rheumatol* 1988; 15: 1833–1840.
35. *node-qrcode – QR code/2d barcode generator*. Available from: <https://github.com/soldair/node-qrcode>. (accessed 5 June 2022)
36. *Information technology. Automatic identification and data capture techniques. Han Xin Code bar code symbology specification*.
37. Galea VP, et al. Longitudinal changes in patient-reported outcome measures following total hip arthroplasty and predictors of deterioration during follow-up: a seven-year prospective international multicentre study. *Bone Joint J* 2019; 101-b: 768–778.
38. *Svelte Framework*. Available from: <https://svelte.dev/>.
39. Mardan A. In: Mardan A (eds) *Using express.js to create node.js web apps, in practical node.js: building real-world scalable web apps*. Berkeley, CA: Apress, 2018, pp.51–87.
40. Vathy-Fogarassy Á and Huguák T. Uniform data access platform for SQL and NoSQL database systems. *Inf Syst* 2017; 69: 93–105.
41. Mardan A. In: Mardan A (eds) *Boosting node.js and MongoDB with mongoose, in practical node.js: building real-world scalable web apps*. Berkeley, CA: Apress, 2018, pp.239–276.
42. Hart SG. NASA-task load index (NASA-TLX); 20 years later. *Proc Human Factors Ergonom Soc Ann Meet* 2006; 50: 904–908.
43. Reichheld FF. The one number you need to grow. *Harv Bus Rev* 2003; 81: 46–54.
44. Roos EM, et al. Knee Injury and Osteoarthritis Outcome Score (KOOS)-development of a self-administered outcome measure. *J Orthop Sports Phys Ther* 1998; 28: 88–96.
45. Ahmad FA, et al. Using REDCap and Apple ResearchKit to integrate patient questionnaires and clinical decision support into the electronic health record to improve sexually

- transmitted infection testing in the emergency department. *J Am Med Inform Assoc* 2020; 27: 265–273.
46. Pellizzoni L, Silva SdAE and Falavigna A. Multilanguage health record database focused on the active follow-up of patients and adaptable for patient-reported outcomes and clinical research design. *Int J Med Inform* 2020; 135: 104065.
  47. Saripalle R, Runyan C and Russell M. Using HL7 FHIR to achieve interoperability in patient health record. *J Biomed Inform* 2019; 94: 103188.
  48. Gray CS. Seeking meaningful innovation: lessons learned developing, evaluating, and implementing the electronic patient-reported outcome tool. *J Med Internet Res* 2020; 22(7): e17987.
  49. Jim HSL, et al. Innovations in research and clinical care using patient-generated health data. *CA Cancer J Clin* 2020; 70: 182–199.
  50. Ploner N and Prokosch H-U. Integrating a secure and generic mobile app for patient reported outcome acquisition into an EHR infrastructure based on FHIR resources. *Stud Health Technol Inform* 2020; 270: 991–995.
  51. Das A, Faxvaag A and Svanæs D. The impact of an eHealth portal on health care professionals' interaction with patients: qualitative study. *J Med Internet Res* 2015; 17(11): e267.
  52. Stover AM, et al. Using an implementation science approach to implement and evaluate patient-reported outcome measures (PROM) initiatives in routine care settings. *Qual Life Res* 2020; 30: 3015–3033.
  53. Barnett GO, et al. Overcoming information overload: an information system for the primary care physician. *Stud Health Technol Inform* 2004; 107: 273–276.
  54. McAlearney AS, et al. Patients' perceptions of portal use across care settings: qualitative study. *J Med Internet Res* 2019; 21: e13126.
  55. Patel PD, et al. Rapid development of telehealth capabilities within pediatric patient portal infrastructure for COVID-19 care: barriers, solutions, results. *J Am Med Inform Assoc: JAMIA* 2020; 27: 1116–1120.
  56. *Breach Portal: Notice to the Secretary of HHS Breach of Unsecured Protected Health Information*. U.S. Department of Health & Human Services – Office for Civil Rights. Available from: [https://ocrportal.hhs.gov/ocr/breach/breach\\_report.jsf](https://ocrportal.hhs.gov/ocr/breach/breach_report.jsf).
  57. Pomputius AF. A review of two-factor authentication: suggested security effort moves to mandatory. *Med Ref Serv Q* 2018; 37: 397–402.
  58. Colnago J, et al. "It's not actually that horrible": Exploring Adoption of Two-Factor Authentication at a University. Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 2018.
  59. Ahn Lv, et al. CAPTCHA: Using Hard AI Problems for Security. in *EUROCRYPT*. 2003.
  60. Patel V, et al. Prescription tablets in the digital age: a cross-sectional study exploring patient and physician attitudes toward the use of tablets for clinic-based personalized health care information exchange. *JMIR Res Protoc* 2015; 4(4): e116.
  61. Mandl KD and Kohane IS. No small change for the health information economy. *N Engl J Med* 2009; 360: 1278–1281.
  62. Sayeed R, Gottlieb D and Mandl KD. SMART markers: collecting patient-generated health data as a standardized property of health information technology. *NPJ Digit Med* 2020; 3: 9.
  63. Sim I. Mobile devices and health. *N Engl J Med* 2019; 381: 956–968.
  64. Dameff C, Clay BJ and Longhurst CA. Personal health records: more promising in the smartphone era? *JAMA* 2019; 321: 339–340.
-