Addressing Chronic Pain Through Patient- and Clinician-Facing Clinical Decision Support: Technical Implementation Guide

Prepared for
Agency for Healthcare Research and Quality
5600 Fishers Lane
Rockville, MD 20857

Prepared by

MedStar Health
Kristen Miller, DrPH, CPPS
Aaron Zachary Hettinger, MD, MS
Raj Ratwani, PhD
Jim Houston, MD
Elias Shaya, MD
Peter Basch, MD
Bonnie Levin, PharmD, MBA, FASHP
Kathryn Walker, PharmD
Ella Franklin, MSN, RN, EDAC
Joseph Blumenthal
Shrey Mathur, MS
Shrenik Shah, MS
John Erkus, MS
Peter Kuehl, MD
Sadaf Kazi, PhD
Nawar Shara, PhD
Ronald Romero Barrientos
Robin Littlejohn, MS
Chris Washington, MA
Chris Bonk, MSN, RN, CNL
Christian Boxley
Deanna Busog

Expert Consultants
Alan Staples, II, CRCR
Ross Teague, PhD
Ranit Mishori, MD, MHS

Vessel Partners
Zach McGill
Doug DeBold

Georgetown University Medical Center
Suzanne O'Neil, PhD, MA
Arnold Potosky, MHS, PhD

George Washington University
Brian Choi, MD, MBA

IMPAQ
Anita Makkenchery, MD, MPH
Kendall Hall, MD
Karishma Desai, PhD

AHRQ Publication No. 24(23)-0043
March 2024

Disclaimer of Conflict of Interest

None of the investigators has any affiliations or financial involvement that conflict with the material presented in this report.

Funding Statement

This project was funded under contract number 75P00119F37007 from the Agency for Healthcare Research and Quality (AHRQ), U.S. Department of Health and Human Services (HHS). This work was supported by the Office of the Secretary Patient Centered Outcomes Research Trust Fund (OS-PCORTF). The opinions expressed in this document are those of the authors and do not reflect the official position of AHRQ or HHS.

Suggested Citation

Clinical Decision Support for Chronic Pain Management. Prepared under Contract No. 75P00119F37007. AHRQ Publication No. 24(23)-0043. Rockville, MD: Agency for Healthcare Research and Quality; March 2024.

Table of Contents

How to Read This Document

What is the Technology and What is it Going to Help Us Do?

The **T**apering **A**nd **P**atient **R**eported Outcomes for **C**hronic **P**ain **M**anagement (TAPR-CPM) application (app) described in this implementation guide (IG) is designed to aid clinicians in designing an opioid tapering plan for patients and to encourage patients to play an active role in reporting their experience and aid in patient-clinician decision making. The technology described in this IG is designed around interoperability standards with a goal to reduce the work needed to implement this electronic health record (EHR)-embedded app.

How Does the Technology Work Together With My System?

The technology described in this IG uses the Substitutable Medical Application Reusable Technology (SMART) security standard and Fast Healthcare Interoperability Resource (FHIR) app coding standard for the clinician-facing app. These technologies enable the clinician-facing app to fit into clinical workflow within the EHR and allows the clinician to use their clinical username/password to access the patient record and plan the opioid taper and treatment plan from within the EHR. The patient-facing app allows the patient to record their own experience using validated measures. These experiences will then be available to the clinicians on the patient's return visit to the clinic and will provide a structured framework for the patient and clinician to make a shared and informed decision about the next steps in the taper and make necessary adjustments.

Key Implementation Decisions

- **Scope of Project:** This app is only focused on tapering opioids and does not include computerized provider order entry (CPOE), custom document creation, medication reconciliation, or other features that are embedded in most EHRs.

- **Patient-Reported Outcomes (PROs):** The patient-facing app uses web app technology to provide a seamless interface on smartphone/tablets, as well as computer browser-based access.

- **Dedicated Server**: The TAPR-CPM app requires a dedicated server to store the PROs, as well as log information necessary to use the clinician-facing app. Attempting to implement the app without a server would significantly limit the functionality of the tool and may prevent the patient-facing tool from working.

- **Authentication Decisions:** The TAPR-CPM app will require an authentication framework for clinicians to access the app in the EHR and for the patients to access their tapering plan and medications in the patient-facing app. Leveraging standard security authentication protocols like SMART and Open Authorization (OAuth) will reduce development time. In addition, using internal health system tools like a patient portal for authenticating a patient before logging into the patient-facing app can make a more secure app that reduces redundancy in the health system's infrastructure.

Introduction

Over the last 20 years, the prescribing of opioid medications for chronic pain has created a large population of patients who have been on long-term opioid therapy, sometimes for decades. Recent therapeutic guidelines from the Centers for Disease Control and Prevention (CDC) encourage clinicians (e.g., physicians, physician assistants, nurse practitioners, and other prescribers) to initiate the gradual deprescribing of opioid medications for select patients in a process called tapering. The process of tapering from opioid medications is challenging for both clinicians and patients. To address these challenges, the Agency for Healthcare Research and Quality (AHRQ) has sponsored the development of clinical decision support (CDS) tools for patients and clinicians to support them through the tapering process. This Technical IG describes considerations for implementing CDS apps that are intended to aid non-pain management specialists in primary care settings to optimize pain therapy and support opioid tapering. The CDS apps are referred to in this technical implementation guide collectively as the **T**apering **A**nd **P**atient **R**eported Outcomes for **C**hronic **P**ain **M**anagement (TAPR-CPM) application (app). The CDS apps are interoperable and publicly shareable to facilitate implementation.

Purpose

This Technical IG provides information and guidance for individuals responsible for the technical implementation of clinician and patient CDS for opioid medication tapering known as the TAPR-CPM app. This guide outlines the standards and technologies used to build the CDS apps, provides strategies and suggestions for the technical implementation of the TAPR-CPM app, and outlines the system requirements for the solutions.

Target Audience

The anticipated primary users of this guide are:

- Healthcare Systems
- Clinicians and Quality Leaders
- CDS Developers and Informaticists

A companion Practice IG is available to assist clinicians with integrating the TAPR-CPM app into their clinical workflow.

Implementing and Using the TAPR-CPM App

Description of the TAPR-CPM

The TAPR-CPM apps are designed to support clinicians and their patients through the process of opioid medication tapering. The key design requirements for TAPR-CPM apps are summarized below.

Clinician-Facing TAPR-CPM App

The clinician app provides an integrated CDS interface within the EHR with the following features:

- Summary of the patient's opioid medications and other medical history.
- Automatic calculations of the oral morphine milligram equivalents (MME) and tapering schedule.
- Generation of an opioid tapering plan snapshot that can be reviewed during future visits.
- Recommendations for proactively treating pain with non-opioid alternatives and symptomatic relief of opioid withdrawal symptoms.

Patient-Facing TAPR-CPM App

The patient app provides a mobile or web-based interface with the following features:

- Collects information from the patient, including Patient-Reported Outcome Measurement Information System (PROMIS) on pain intensity, pain interference, and daily pain.
- Provides within-app (or links to) patient education on non-opioid related pain management strategies.
- Provides motivation, coaching, and feedback on progress through the tapering process.
- Provides information to patients on the details of their taper plan.

Overview of Architecture

Figure 1 outlines the data flow and architecture of the TAPR-CPM app. The clinician app has a bi-directional transmission of data to the data hub and external assessment center. The app sends created taper plans to the hub and ingests self-reported patient data captured by the data hub from the patient app. The patient and clinician apps will also query the EHR FHIR endpoints for patient data, while being authenticated against the EHR's firewall. The net scaler artifact will provide load balancing for requests to serve the clinician and patient apps. Load balancing distributes these requests across the resources of the application in order to make sure the application continues to run efficiently and reliably.

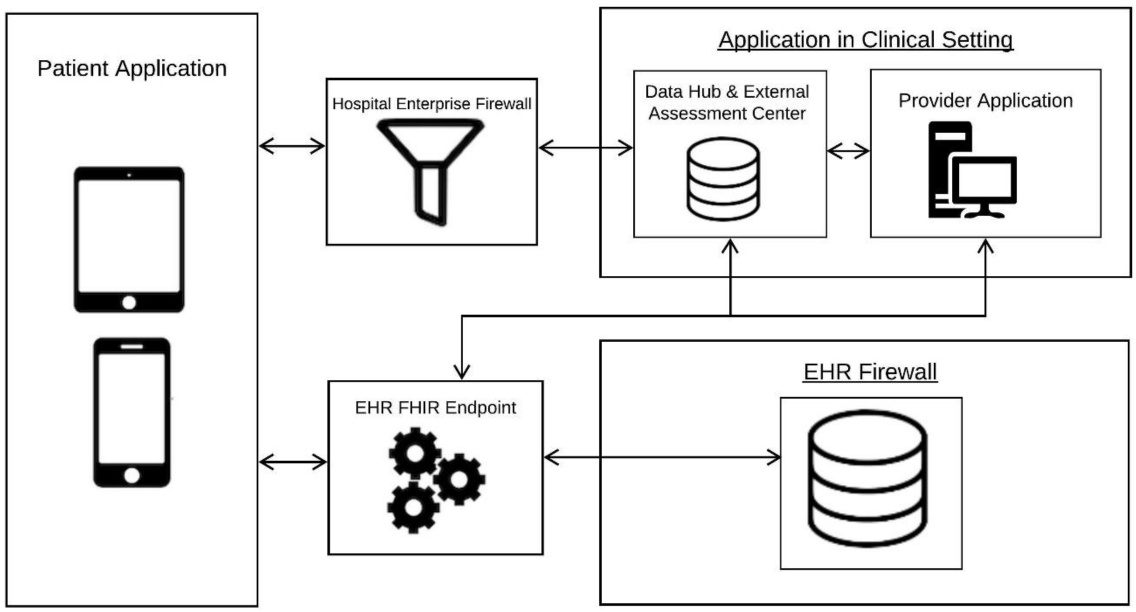

Figure 1. Overall TAPR-CPM architecture

Design Decisions and Implementation Considerations

The TAPR-CPM app for patient and clinician CDS for opioid tapering was designed using standards for interoperability to reduce development time across as many EHR solutions as possible. Descriptions and sources for the standards and technologies used are summarized in Appendix C. This section of the guide briefly summarizes the design decisions and the impact of those decisions on the implementation of the TAPR-CPM apps within your local context.

Design Decision: The TAPR-CPM App Must be Built Within Current Standards for Interoperability

Interoperable CDS tools built with current standards are beneficial because of their shareable nature and decreased emphasis on local customization. This “plug and play” nature was a desired feature for the build of the apps. The interoperability standards and technologies considered when building the patient and clinician TAPR-CPM apps include Health Level-7 (HL7) FHIR standards and SMART technology where applicable. Additional standards not discussed in depth include the use of RxNorm, International Classification of Diseases Tenth Revision (ICD-10), and others for coding the data contained in the EHR. The use of these standards and associated value sets from the National Library of Medicine's Value Set Authority Center are critical to the interoperability of the TAPR-CPM app and reducing development and implementation timelines.

- National Library of Medicine Value Set Authority Center

Design Decision: Leverage FHIR Technology

FHIR is a data standard that aims to provide technical specification and documentation concerning the formatting, standardizing, and electronic exchange of healthcare data. Part of the value in using FHIR is that it works to standardize data across different EHRs and labels these data elements as resources.

The TAPR-CPM app parses and reads a patient's current medication data using FHIR resources. However, differences in the FHIR resources between different versions of the FHIR specifications/standards may result in a mismatch of information shared from the patient-facing app into the clinician-facing app. For example, in the DSTU2 FHIR specification (Version 2), we identified relevant patient medication data to be contained in the "Medication," "MedicationOrder," and "MedicationStatement" resources. In the most recent FHIR specification, R4 (Version 4), relevant patient medication data is housed under the new "MedicationRequest" and "MedicationAdministration" resources, which, along with the "MedicationStatement" resource, should provide the relevant patient data for the apps to use.

Different EHR vendors and health information technology (IT) systems in the healthcare organization may only be compatible with a specific version of FHIR. As a result, the clinician-facing tool utilizes FHIR R4 but the TAPR-CPM Data Hub (see below for details) converts and stores the data in the FHIR DSTU2 format to allow for the broadest compatibility.

Implementation Consideration: Organizations looking to implement the TAPR-CPM app should investigate the current FHIR specification in use within their EHR and other health IT components that the app may utilize.

Design Decision: Leverage Security Standards

SMART is a standards-based platform that allows for the development of interchangeable healthcare apps that draw from and communicate with any EHR data in a healthcare system. SMART enables healthcare developers to reuse their developed technologies across different EHR platforms, which results in increased interoperability.

For the TAPR-CPM app, the development team implemented SMART standards that are a part of the SMART on FHIR app authorization protocol to secure the exchange of sensitive patient information throughout the network of the apps (e.g., EHR, TAPR-CPM). Transport Layer Security (TLS 1.2) is used to encrypt data being communicated through the proposed app architecture and data flow. Internet protocol open standard for access delegation, OAuth 2.0 will be used by the solution's servers for user authorization and to establish identity and granting of JavaScript Object Notation (JSON) web tokens to allow the apps to read and parse through sensitive patient data in a protected manner. Because the app will consist of two end users—the clinician and the patient—further authentication will take place for both apps. Because the clinician-facing app will be hosted within a health system's EHR instance, the clinician wishing to access the app will be authenticated upon EHR launch by the health system's credentialing and role assignment procedure. The standalone patient app will be authenticated using the patient portal with OpenID Connect being leveraged to collect identity data from the patient app to partition the appropriate levels of data access to users.

Implementation Consideration: Secure implementation of the TAPR-CPM apps will require organizations to use existing security standards. The TAPR-CPM tools leverage SMART along with TLS 1.2 to secure data. For user identification, the apps use OAuth 2.0 and OpenID Connect to work with a healthcare organization's existing solutions, like EHR user validation and the patient portal. Alternatives to these standards may be less secure or more challenging to implement with a vendor's EHR solution.

Design Decision: External Assessment Center and Patient-Reported Outcomes

A key function of the patient-facing TAPR-CPM app is the administration of PROMIS measures. PROMIS administers patient-reported outcomes (PROs), which are patient-generated health reports that come directly from the patient without interpretation by a clinician or health professional. To enable this functionality, we leverage two existing IGs: the Office of the National Coordinator for Health Information Technology (ONC) Structured Data Capture (SDC) IG and the Patient-Reported Outcomes IG.

- ONC SDC IG
- Patient-Reported Outcomes IG

The SDC IG is an extension to the FHIR resources "Questionnaire" and "QuestionnaireResponse" that enables significantly more complex data and interactions than the basic Questionnaire. The PRO IG extends the SDC IG and standardizes more concepts specific to the use case of capturing PRO measures using FHIR. Among the key concepts that the PRO IG defines is an external assessment center (EAC). From the IG: "The [EAC] is a system that is capable of administering a questionnaire based on item response theory (IRT) algorithms." Additionally, the EAC does not need any information about the specific patient.

The TAPR-CPM leverages PROMIS measures and the EAC. This EAC conforms to both the SDC IG and the PRO IG.

- PROMIS Measures
- External Assessment Center

Implementation Consideration: By conforming to existing IGs (e.g., SDC IG, PRO IG), a healthcare organization could choose to use a different EAC or build their own EAC. The system is configured such that switching to a different EAC is a straightforward process and does not lock future implementation into using the EAC leveraged in this IG.

Design Decision: Computer Adaptive Tests

A key challenge in any patient-facing app is designing the app to increase usage and compliance with what has been asked of the patient. Computers and mobile apps allow us to leverage computer adaptive testing (CAT) for the administration of the PRO measures in the patient-facing TAPR-CPM. When administering CAT-based PROs, a computer uses a validated algorithm to determine which question is going to collect the most information about a patient's condition and then displays that question to the patient. This algorithm ensures that only the smallest number and most relevant questions will have to be answered by the patient.

- CAT

Implementation Consideration: To reduce complexity, organizations could also choose to have the patients answer all PRO items within a measure at the expense of added time for the patient, reduced app usability, and potential impact on the completeness of the data received.

Design Decision: Computing Resources

The TAPR-CPM app leverages cloud computing resources to avoid the large, upfront expenses of purchasing servers to host the app. By developing cloud native software, we gain several important operational efficiencies. First, we can purchase only the computer processing (compute) resources that are needed at the time, saving money. Second, as a solution grows and scales, adding more compute resources can be accomplished quickly and easily. Third, these compute resources are not specific to any one cloud provider, allowing the shift to different cloud provider if the current cloud provider is not the best option for the project.

Implementation Consideration: Healthcare organizations can choose to host the app on their own server resources or choose to leverage cloud-based infrastructure as described in this implementation guide.

This project utilized hosted Linux and Windows cloud instances to deliver the compute resources that were needed to run the application programming interfaces (APIs), web, and mobile apps. The decision to use one cloud provider over another did not have to do with technical capabilities, but with the preference of the initial pilot site. Any cloud service provider could have been leveraged, and future implementors are free to choose their own cloud provider or even host the apps on their own institutional hardware with minimal extra configuration necessary.

Implementation Consideration: Any cloud-based company should be able to provide the necessary infrastructure for the development of this app, and choosing a cloud provider should be based on existing practices in the healthcare organization.

Design Decision: Server Roles

The TAPR-CPM apps have three logical partitions within the backend. Any logical partition can be hosted either on-premise or on a cloud server. The following sections describe each logical partition and proposed server roles. Please note that the final version of the code and logic models will be included in the CDS connect artifact.

Logical Partition 1 - App Delivery Framework

Services contained in front-end app delivery should be responsible for communicating with end-user devices such as mobile devices, desktops, and devices used for getting alerts such as short message service (SMS) or emails.

Patient-facing and clinician-facing TAPR-CPM app deliveries should be handled by web apps running on a secured app server. Each or both apps can be served by one web server or by multiple, load-balanced servers depending on the app load and availability requirements. Both apps should use a standard web delivery mechanism to deliver user interface content to respective devices using secured web data transfer protocols (hypertext transfer protocol secure—HTTPs / Transport Layer Security—TLS).

Implementation Consideration: The healthcare organization may use any third-party service to deliver SMS or email messages; however, some services do not offer Health Insurance Portability and Accountability Act (HIPAA)-compliant services. Special care should be made to how protected information is transmitted.

Logical Partition 2 - Data Persistence

Services contained in the data persistence layer should be responsible for:

- Storing data and the configuration information necessary for proper working of the entire TAPR-CPM app.
- Storing data generated by virtue of user interaction with either the patient-facing or clinician-facing applications by storing it in a database server so data in transit as well as at rest remains encrypted.
- Implementing mechanisms so that logging and app configuration data gets stored on respective app servers as log and configuration files.
    - Log or configuration data should not have any patient identifiers and, when possible, should only be accessible from within the server to users authorized with access.

Logical Partition 3 - App Logic

Services contained in the app logic partition should be responsible for implementing FHIR standards for data communication, connecting to and implementing the EAC for handling questionnaire-responses, administering PROs using the CAT framework, connecting to RxNorm API, and implementing oral MME and dosage calculations as needed. These services should use secured transport (HTTPS / TLS) for all data transfer.

Logical to Physical Server Mapping

All the logical partitions can be implemented in one or more physical, on-premises virtual, or a cloud server, depending on your organization's information technology service standards.

Implementation Plan

Brief Description of the App in Clinical Settings

The proposed high-level workflow is depicted in Figure 2. The clinical workflow is divided into three primary functions:

1. Initial visit workflow
2. Followup visit workflow
3. Patient onboarding workflow

A detailed proposed workflow for clinicians is available in Appendix D of the Practice IG.

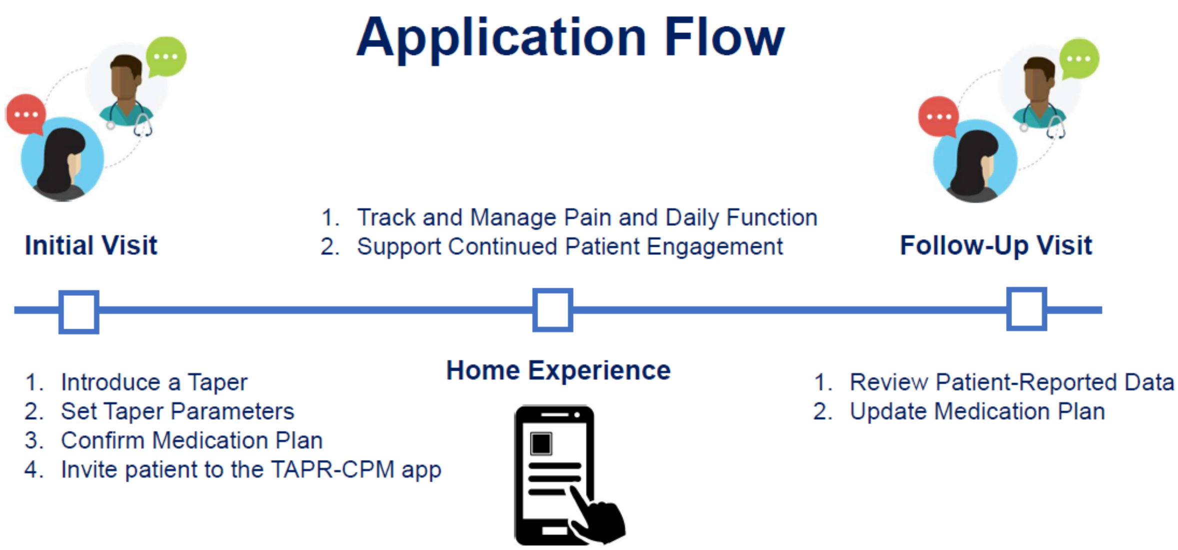

Figure 2. High-level clinical workflow

Clinician Workflow for Initial Visits

The workflow for creating a taper plan is guided by the clinician-facing TAPR-CPM app and is presented as a series of sequential screens during the initial patient visit (see Appendix H of the Practice IG). After the taper plan has been created, the suggested workflow is for the clinician to trigger an invitation link to be delivered to the patient through text message or email. This invitation is done directly through the clinician-facing TAPR-CPM app. After sending the link, it is suggested the clinician copy the clinician summary section into the patient's visit notes within the EHR. The clinician may also print the patient summary information for the patient's reference.

Clinician Workflow for Followup Visits

The suggested workflow for followup visits is for the clinician to review the patient-generated data available within the clinician-facing TAPR-CPM app either before or during the patient's return visit. At this point, the clinician and patient may decide to continue the taper plan at the current settings or to make a change. The clinician must then create the next taper interval accordingly within the clinician app. Lastly, the clinician can again copy the summary into their notes within the EHR and print the patient summary.

Patient Onboarding Workflow

For patients deemed appropriate by clinicians for tapering and use of the patient-facing TAPR-CPM app, patient onboarding is a critical workflow. It is highly suggested the patient is sent the invitation link to their email or smartphone device by the clinician during the initial visit. If this step is not possible, we recommend having a staff member facilitate this process. Once the patient receives the link, they will not be able to access the app until they sign in using their patient portal username and password. We recommend that the clinician or staff ensure the patient has the patient portal credentials available to them. Any patient that does not have access to their patient portal credentials should be appropriately directed to create a patient portal account. Once patients have signed in, the app will lead them through accepting the terms of use, data authorizations, and completion of the patient-reported outcome surveys.

EHR FHIR Endpoint

The FHIR endpoints provide a reliable, secure authorization protocol for a variety of app architectures. Apps that need to access FHIR resources by requesting access tokens from OAuth 2.0-compliant authorization servers use FHIR endpoints. OAuth 2.0 is compatible with FHIR DSTU2 and above. The OAuth 2.0 framework defines a mechanism for a client application to request access to protected resources based on a set of scopes, and then have that access granted or denied by the user and the server. The Smart on FHIR standard defines several scopes, as well as a general method for requesting access to any FHIR resource.

- Smart on FHIR

Design Decision: Data Hub

To enable interoperability and portability, the TAPR-CPM app leverages the HL7 FHIR R4 standard to store and transmit data. This standard allows the app to easily and quickly access the necessary data that are stored in the EHR, like patient data and medication data for patients. EHRs are currently limited in the FHIR data types that can be retrieved from them, and even more limited in what data can be written back and stored. For example, at the time of writing, we are unaware of any major EHR that supports the Questionnaire or QuestionnaireResponse resources, both of which are required for administering PROs. To address the issue of not being able to store and retrieve important data types, such as “Questionnaire” and “QuestionnaireResponse,” the TAPR-CPM app features an architecture that leverages an external FHIR server in addition to the EHR’s FHIR APIs. All FHIR resource types where write capabilities are needed will be stored in this external server, called the Data Hub (Figure 1). Examples of resources that need to be recorded include the “QuestionnaireResponses” resource for the PROs and the clinician-generated opioid taper plan as a FHIR “CarePlan.” To accomplish this link between the EHR and the Data Hub, the TAPR-CPM app uses FHIR external resource references to maintain a link back to the identity of the patient in the EHR, while allowing access to the needed data.

- HL7 FHIR R4 standard

Implementation Consideration: The healthcare organization may enable direct writing to the EHR of clinical documents, medication orders (prescriptions), and other data elements. While this option will increase the usability of the app, it also increases potential risk of error, development complexity, and implementation timeline.

Clinician-Facing TAPR-CPM App

The clinician-facing app is designed to provide both basic functionality and a tool that helps both the clinician and the patient to effectively taper opioid medications. The clinician app user interface (UI) is divided into two parts: an initial visit dashboard and a followup dashboard.

Initial Visit Dashboard

The pre-visit app dashboard is divided into five pages, which are all built using hypertext markup language (HTML), jQuery, cascading style sheets (CSS), JavaScript, and D3.js (see Appendix C for descriptions of these technologies).

Page 1: Patient Context and Current Medications

The first page displays the current opioid medications that the patient is taking, along with the prescription drug monitoring program (PDMP), other medications, social history, currently active problems, and all diagnoses. The first page also provides certain steps and help guidelines to support the clinician in optimizing the use of the app by providing step-by-step instructions on app use. In addition, Page 1 provides links to the evidence-based tapering guidelines from the CDC and the Department of Veterans Affairs (VA).

Page 2: Taper Settings and Speed of Taper

Page 2 displays the available tapering speeds and allows the selection of the start date. Oral morphine MMEs are an opioid dosage equivalency to oral morphine, and it is the metric we are using as a gauge of the potency of the amount of opioid that is being given to a patient at a particular time. To help visualize the taper, there is an oral MME visualization graphic that compares the current opioid medication dose with the future tapered dose. The time to 100% taper is also provided.

Page 3: Opioid Taper Plan

Page 3 displays a summary of the opioid medication information to the clinician, along with the progress the patient has made on the taper plan.

Page 4: Non-Opioid Taper Plan

Page 4 displays information for tapering of non-opioid medications, along with pertinent laboratory results (liver and kidney). Page 4 also provides alternate methods and activities in which the patient can reduce pain, such as yoga, physical therapy, and acupuncture.

Page-5: Clinician Summary

Page 5 includes a clinician and patient summary. This summary includes the change in the opioid medications according to the taper plan and an option to add the changes to the visit notes.

Followup Visit Dashboard

The followup dashboard consists of Pages 1-5 with one important distinction: if the patient has completed PROs in the patient-facing apps (described below), these will be displayed along with oral MME calculations and a graphical representation of temporal changes to the patient's measures and opioid prescriptions. The subsequent screens will include all of the same information observed in the initial visit dashboard.

Patient-Facing TAPR-CPM App

The patient-facing TAPR-CPM app may be accessed either via the web or through a downloadable mobile app. The iPhone Operating System (iOS) and Android apps are designed as Hybrid Mobile Apps. To enable this hybrid mobile app architecture, we chose to use Ionic + Capacitor, two open-source projects created by the Ionic Framework company. Ionic + Capacitor takes the web code that is written and wraps native iOS and Android wrappers around it to allow it to run as a native app. Details will be available in the CDS Connect artifact.

- Hybrid Mobile Apps

To build the patient-facing app, we did a broad search of the best tools and platforms to use. We selected React, an open-source framework made by Facebook, as the foundation of our app. On top of React, we chose to use Material-UI (https://material-ui.com/), a React-specific implementation of Google's Material Design standard. Material-UI provides many common web components like buttons, links, and forms as pre-built components. These pre-built components were adapted to our design system and exposed as a library that we can use to rapidly build out new app features. This library provides a balance of well-built and tested components, and the ability to modify the components to match the design of the app. These components will be made available via the CDS Connect artifact.

- React Framework

Implementation Consideration: The patient-facing app uses modern web-based programming frameworks (including React and Material-UI) that may be less familiar to healthcare organizations. These provide faster development time and consistency of experience across patient devices. Healthcare organizations may decide to convert the app to a more familiar language.

App Integration

To guarantee a smooth integration and implementation of the solutions across the healthcare organization and anticipated end users, it is important to consider the different integration points that are needed to enable full functionality of the clinician- and patient-facing apps. In this section, we will outline the integration points of the app, including implementation of the SMART standard, integration into the EHR, patient-facing integrations, and integration of the data hub across both apps.

Implementation of SMART Standard

A part of the SMART standard, the SMART App Launch Framework connects third-party apps to EHR data, allowing apps to launch from inside or outside the user interface of an EHR system. The app is responsible for protecting itself from possible faults or malicious values passed to its redirect uniform resource locator (URL) (e.g., values injected with executable code, such as Structured Query Language—SQL) and for protecting authorization codes, access tokens, and refresh tokens from unauthorized access and use. Before a SMART app can run against an EHR, the app must be registered with the system's EHR authorization service. OAuth 2.0 authorization servers are configured to mediate access based on a set of rules configured to enforce institutional policy, which may include requesting end-user authorization. Scopes, Access Tokens, and Refresh Tokens are some of the security features that have been built into the OAuth 2.0 protocol that have been leveraged to mitigate attacks and security issues. The clinician-facing app uses the SMART launch specification for the EHR launch sequence.

- SMART launch specification

Clinician-Facing EHR Integration

While the clinician-facing app's functionality relies on user-generated data from the clinician and patient-facing mobile app, it also leverages heterogenous data sources from multiple artifacts in the architecture. When designing an app to ingest data from multiple sources, the control flow of these inputs and outputs are an important consideration as they impact code maintenance and app performance.

One design consideration is whether the code can be written and supported using modern web standards to optimize performance to avoid serialization of requests that increase latency of UI load time. It is best practice to run requests for data retrieval and storage in parallel. This method of asynchronous data retrieval requires deliberate code design patterns to avoid unintended consequences if a request for data is unsuccessful, and to provide readability and decrease complexity of the code. Many of these libraries (e.g., jQuery functions like “deferred” and “promises”) provide built-in functionality to add flexibility in managing the requesting and error recovery and are commonly used in industry, but not in healthcare systems development. Most web browsers natively support the functionality of managing requests; however, the functionality need to be included in EHR by using backward-compatible libraries such as Bluebird.js.

The SMART launch framework described above is used when connecting applications to the EHR endpoints themselves, and when connecting to non-EHR endpoints, additional options are available. The recommended choice is to use the encrypted signed token issued by the EHR authorization server during the clinician-facing EHR launch sequence. This approach has two distinct advantages. The first advantage is that the EHR's public key for decryption is freely available to authenticate the identity of the issuer and, therefore, using the EHR itself for clinician identity. The second advantage of this approach is that the authorization server with which you are handshaking and completing the authorization process has additional information, including the medical system’s internal identifier for the patient for whom the application is being launched.

For integration with artifacts implemented at a system level, such as the prescription drug monitoring program (PDMP), it may be advantageous to leverage an existing architecture. PDMP programs may exist in isolation or be integrated into local health information exchange (HIE) systems. HIEs typically establish trust between health systems and the central repository of shared data, which makes integration of individual apps to the HIE directly a challenging approach. The optimal solution may be to use the same authentication and communication channels as other client apps inside the health system, thereby reducing the effort for integration of PDMP data into the clinician application.

Implementation Consideration: The integration of the PDMP in the clinician-facing opioid tapering app provides critical information and increased usability but may not be feasible if the PDMP is not currently integrated in the organization’s EHR and requires logging into external HIE to review.

Patient-Facing TAPR-CPM App Integration

During the design phase of integration of the patient-facing app, several considerations allow for a better user experience. One key decision point in an authentication flow is to verify the user’s identity, known as an identity provider (IDP) for the patient when signing into the app. Though there is a wide selection of IDPs used in the industry, a consideration for the implementation of the TAPR-CPM app is to use the patient portal of the health system. One potential limitation of using the patient portal as the IDP is that the patient must already have an account with the health system and therefore would be unable to enroll otherwise. The advantages include the potential to save the patient from needing to create another set of credentials, as well as the direct issuance of an access token to exchange for EHR data. The decision of whether or not to leverage the patient portal account or a separate registration system may depend on its availability from the EHR vendor, the app or health system's registration process, or the workflow of enrolling the patient to use the app itself.

Implementation Consideration: The healthcare organization will need to select an IDP to utilize for patient identification authentication for the patient-facing TAPR-CPM app account. Ideally, a solution that is already in place can be selected to provide this important security component.

Data Hub Integration

As the bridge between the patient-facing and clinician-facing TAPR-CPM apps, the data hub has the greatest number of integration points. As such, it is good practice to test these integration points as early in the implementation and integration process as possible. This approach has several advantages. The first advantage is the opportunity to iterate and fix any connectivity issues early in the process. As the hub is the central spoke of integration between the patient- and clinician-facing TAPR-CPM apps, it may be beneficial to build APIs for testing using staged data. This testing allows for the clinician- and patient-facing apps to begin to test "the plumbing" and begin to ingest data into the apps for development.

Another design decision is whether or not a health system wants to invest in the upfront cost to add a layer of authentication to uniquely identify the user of the patient app. While there is an associated additional development cost, and it is not required by the SMART launch sequence documentation, it is recommended that the hub only allow connections from the patient-facing TAPR-CPM app when an OpenID connect is available. Although OAuth 2 allows for secure delegation of access to resources, it does not uniquely identify the patient. OpenID connect is an assertion of identity that allows the hub to differentiate between a proxy account holder (e.g., caregiver with delegated access) from the patient themselves. The designated proxy in the patient's portal may allow a proxy to see lab results, for example, but should not be allowed to see the patient's pain journal unless approved.

Systems Requirements and Development Documents

A key to successful implementation is the planning and documentation of requirements. The system requirements document (SRD) and system design document (SDD) provide the opportunity to socialize and document requirements from stakeholders and detail a plan for its execution. The high-level goal of these documents is to codify the precise goals of the system to address and outline a subsequent tailored plan for implementation. Creating such documents, however, provides a trade-off for both the health system and development team. These documents are detailed and time intensive to create, review, and finalize. If a health system prefers to begin integration and user testing sooner, it may decide to forgo these detailed requirements and their documentation. If a health system decides they would like to invest the upfront time to create these documents tailored to their system, they may be able to minimize the chances that components may not function exactly as users expect and meet their needs.

The SRD and SDD often include a list of stakeholder-defined requirements from interviews and details collected from direct discovery of user needs. The requirements document can be outlined and identified by unique numbers for subsequent reference and references a specific stakeholder group to tie these numbers to the user need. This document is meant to be created in collaboration with each stakeholder group and allows socialization of the solution's functionality prior to outlining the steps needed to achieve those goals. After each of the system's requirements are outlined, it can be advantageous to tailor an implementation plan with specific goals, tests, and acceptance criteria, which refers directly to each identified user need listed in the SRD/SDD.

Two dimensions of planning and testing of note would be both unit and module testing and their associated acceptance criteria. Unit testing typically addresses the method of validation of individual components built to address each need outlined in the SRD/SDD. The suggested method of outlining the unit testing is to directly refer to the objective, scope and principles, tasks to be performed, and acceptance criteria for completion. While unit testing can be thought of as testing each individual component, module testing refers to a plan to test how each component integrates and to ensure the system itself functions. As module testing refers to the app's ecosystem, it can be helpful to outline these as integration points for which each of the modules connect and their interdependencies. This testing can be especially important if multiple apps are being built and tested in parallel. It is helpful to outline these test scripts as a one-to-one reference to the requirements defined in the SRD/SDD, as well as the sequence of steps, expected results, and pass criteria.

AHRQ Publishing and Communications Guidelines

Site Integration Plan

Site integration plans are critical to a successful implementation. While there will always be site-specific challenges associated with individual implementations, creating an overall template that can be tailored to health systems can help minimize the risk of delays in technical deployments. One helpful tool is to create a minimum standards requirement document that outlines the base level of readiness for implementation. Some dimensions to consider would be current FHIR standards, approval processes and timelines, and vendor versus local technical support. The following sections provide some considerations for how to approach such a plan to help generalize an implementation plan.

Integration Template for All Sites

One approach to creating a technical implementation plan spanning EHR platforms and sites is to plan for variability by designing the app's architecture and code with flexibility. The architecture itself should be created such that each artifact is able to be implemented separately to decrease dependencies in case of technical challenges. The clinician app UI should design for the lowest common denominator of support from the visualization platform in the EHR. While individual vendors provide various levels of underlying browser support for SMART apps, even different versions of the same EHR can provide diverse levels of support. Unfortunately, lack of support and networking challenges limit developer options for using modern web tools that are readily available and common in web development.

Although FHIR is an interoperable standard for communication, the implementation of these specifications varies widely. A great strength of the FHIR standard is that meta data regarding permitted usage of resources and operations are clearly defined and required to be available for each implementation. This requirement allows the apps to "ask" each FHIR endpoint what resources it implements and their permitted operations. This ability to reference the FHIR endpoints capabilities allows app design to improve efficiency, but the code must be designed to handle cases when these advanced operations are unavailable. To validate this design, the code should be tested using as many publicly available tools and SMART on FHIR sandboxes as possible to plan for variance in implemented functionality.

Anticipated Technical Challenges

The overall value of weighing decision points outlined in the technical sections of the implementation guide will help reduce the likelihood of encountering unanticipated technical challenges. However, each health system, site, and practice have unique challenges associated with technical build and deployment. It is helpful to build in as much extra time as possible to address any unanticipated challenges. While it is known that unanticipated challenges are inevitable, building safe and efficient applications that can effectively adapt their approaches depending on the institution are critical to a successful implementation. One such example is developing the application's logic to be configurable to allow for different coding systems (e.g., conditions may be expressed as Logical Observation Identifiers Names and Codes—LOINC, Systemized Nomenclature of Medicine—Clinical Terms—SNOMED-CT, or both).

Appendix A. Glossary of Terms

Table A1. Glossary of terms

Term	Definition
Clinicians	Providers, physicians, physician assistants, nurse practitioners, and other primary prescribers
Cloud Computing	The practice of using a network of remote servers hosted on the internet to store, manage, and process data, rather than a local server or a personal computer
Compute Resources	An infrastructure service that provides resources in the form of virtual machines
Oral Morphine Milligram Equivalents	Amount of morphine in milligrams that is equivalent to the opioid dose component in a prescription
RxNorm	Provides normalized names for clinical drugs and links its names to many of the drug vocabularies used in pharmacy management and drug interaction software

Appendix B. Acronyms and Abbreviations

Table B1. Acronyms and abbreviations

Acronym / Abbreviation	Definition
AHRQ	Agency for Healthcare Research and Quality
API	Application Programming Interface
App	Application
CAT	Computer Adaptive Testing
CDC	Centers for Disease Control and Prevention
CDS	Clinical Decision Support
CPOE	Computerized Provider Order Entry
CSS	Cascading Style Sheets
DSTU	Draft Standard for Trial Use
DSTU2	Draft Standard for Trial Use 2
EAC	External Assessment Center
EHR	Electronic Health Record
EPLC	Enterprise Lifecycle
FHIR	Fast Healthcare Interoperability Resource
Health IT	Health Information Technology
HHS	Health and Human Services
HIE	Health Information Exchange
HIPAA	Health Insurance Portability and Accountability Act
HL7	Health Level-7
HTML	Hypertext Markup Language
HTTPS	Hypertext Transfer Protocol Secure
ICD-10	International Classification of Diseases, Tenth Revision
IDP	Identity Provider

Acronym / Abbreviation	**Definition**
IG	Implementation Guide
iOS	iPhone Operating System
IRT	Item Response Theory
IT	Information Technology
JSON	JavaScript Object Notation
LOINC	Logical Observation Identifiers Names and Codes
MME	Morphine Milligram Equivalent
OAuth	Open Authorization
ONC	Office of the National Coordinator for Health Information Technology
PDMP	Prescription Drug Monitoring Program
PHR	Personal Health Record
PRO	Patient-reported outcome
PROMIS	Patient-Reported Outcome Measurement Information System
R4	Mixed Normative and STU
SDC	Structured Data Capture
SDD	Systems Design Document
SDLC	System Development Lifecycle
SMART	Substitutable Medical Applications Reusable Technology
SMS	Short Message Service
SNOMED-CT	Systematized Nomenclature of Medicine – Clinical Terms
SQL	Structured Query Language
SRD	Systems Requirement Document
STU3	Standards for Trial Use 3
TAPR-CPM	**T**apering **A**nd **P**atient **R**eported Outcomes for **C**hronic **P**ain **M**anagement

Acronym / Abbreviation	Definition
TLS	Transport Layer Security
UI	User Interface
URL	Uniform Resource Locator
VA	Veterans Affairs

Appendix C. Descriptions of Standards and Technology

Fast Healthcare Interoperability Resource Technology

Fast Healthcare Interoperability Resource (FHIR) is a data standard that aims to provide technical specification and documentation concerning the formatting, standardizing, and electronic exchange of healthcare data. FHIR standards, created by the standard setting organization HL7, are continuously evolving and improved upon with feedback from the healthcare information technology (IT) community. Since 2012, there have been four different versions of FHIR made available by HL7: DSTU1, DSTU2, STU3, and R4, with R4 being the most recent FHIR specification released in December 2018. The TAPR-CPM apps comply with the specifications. Specific FHIR resources used in the apps are listed in Table C1.

Table C1. Summary of FHIR resources required for this project

FHIR Resource	Description	Link
Patient	Demographic information about the patient	http://www.hl7.org/fhir/r4/patient.html
MedicationRequest	Information about requests for medications for patients	http://www.hl7.org/fhir/r4/medicationrequest.html
Observation	Information about observations performed by a clinician	http://www.hl7.org/fhir/r4/observation.html
Condition	Information about a patient's condition	http://www.hl7.org/fhir/r4/condition.html

SMART Standards

SMART is a standards-based platform that allows for the development of interchangeable healthcare apps that draw from and communicate with any EHR data in a healthcare system. By creating open standards to be used for healthcare app development, SMART enables healthcare developers to reuse their developed technologies across different EHR platforms which results in increased interoperability.

- SMART health IT

External Assessment Center and Patient-Reported Outcome Measures

A key function of the patient-facing TAPR-CPM app is the administration of PRO measures. To enable this functionality, we leverage two existing IGs: the ONC SDC IG and the PRO IG.

INC SDC IG

PRO IG

Computer Adaptive Tests

One of the key challenges in any patient-facing app is designing the app in such a way as to increase usage and adherence with what has been asked of the patient. A valuable approach to accomplish this objective is to reduce the time that it takes to complete tasks.

One of the core tasks that users of the patient app will accomplish is completing PRO measures. Traditionally, PROs were long, paper-based questionnaires with many questions that were not necessarily relevant to the patient’s condition. With the advent of computers and mobile apps, a new form of administering PROs was made possible called computer adaptive testing (CAT).

When administering CAT-based PROs, a computer uses an algorithm to determine which question is going to collect the most information about a patient’s condition and then displays that question to the patient. This approach ensures that only the most relevant questions will have to be answered by the patient. By using this approach, the total number of questions that a patient must answer to arrive at a reliable score can be greatly reduced.

Technologies for Clinician-Facing App Dashboard Development

Table C 2. Clinician-facing app dashboard development technologies

Term	Description
HTML	Hypertext Markup Language is the standard markup language and the basic building blocks for documents designed to be displayed in a web browser.
jQuery	A fast, small, and feature-rich JavaScript library, it makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.
CSS	Cascading style sheets describes how HTML elements are to be displayed on screen, paper, or in other media.
JavaScript	A lightweight programming language with first-class functions, it is most well-known scripting language for Web pages. It is a single-threaded, dynamic language, supporting object-oriented, imperative, and declarative (e.g., functional programming) styles.
D3.js	A JavaScript library for manipulating documents based on data. D3 allows you to bind arbitrary data to a document object model (DOM), and then apply data-driven transformations to the document to generate visualizations.

Appendix D. Sample Legal Language for Digital Health Applications

Whether your healthcare system is new to the digital health space, or enhancing its existing digital health app portfolio, important aspects of the tools require legal, risk, or compliance review. Each healthcare system will vary in their level of involvement; however, these are potential issues you should discuss/review with your legal team, and the statements below are meant to give an example and may be neither necessary nor sufficient for your implementation.

Privacy Policy, Terms of Service, and Medical Advice Disclaimer

- **Medical Advice Disclaimer:** [Example] The content and information contained in this app is for informational and reference purposes only and should not be used as a substitute for medical advice, diagnosis, or treatment, or as the sole source of guidance for decision-making. We advise you to always consult with a physician before making any healthcare decisions or for guidance about a specific medical condition. If you are experiencing a medical emergency, call 911 immediately, or go directly to your nearest emergency room.
- **Information Disclaimer:** [Example] While all efforts are made to ensure that the information contained in this app is correct and up to date, the accuracy and completeness of the information is not guaranteed. You should use additional sources to verify information gathered from this app.
- **Disclaimer of Liability:** [Example] Reliance on any content or information provided in this app is solely at your own risk. [PRACTICE NAME] and its affiliates expressly disclaim responsibility, and shall have no liability, whatsoever, for any damages, loss, or injury suffered as a result of your reliance on the content or information contained in this app.
- **Third Party and Advertisement Disclaimer:** [Example] This app may, from time to time, contain links or references to third-party sources of information. These links and references are provided solely as a convenience to you and not as a guarantee, warranty, or endorsement of the services or content provided by such third-party sources. [PRACTICE NAME] and its affiliates are not responsible for the content of third-party sources that are linked to or referenced and do not make any representations or warranties regarding the content or accuracy of information or materials provided by such third parties.
- **Privacy Policy:** [Example] This app collects data solely for the purpose of research and will be permanently deleted upon conclusion of the study. The app

will collect only the following data points: name, date of birth, patient identifier from the medical record, email or cellular phone number, measurement of your physical function measured by a survey you take, and the score of the survey.

Security Considerations

Security considerations will vary based on the specific deployment through a patient portal, secure text messaging, and/or emails through the TAPR-CPM app.

PROMIS Licensing

PROMIS measures are publicly available for use in individual research, clinical practice, educational assessment, or other app without licensing or royalty fees. However, deployment through the TAPR-CPM app leverages the external assessment center (a web-based data collection platform), and the assessment center application programming interface (API)—an API that connects data collection software with the full library of PROMIS measures including the computer adaptive tests (CATs). Licensing requirements will vary by site depending on EHR vendor and current licenses.