

SMART on FHIR: Idea to Implementation

Contents

Topic: Organisational Contexts, Workflows and User Journeys.....	2
How to Document Workflows: Key Elements.....	2
Structure of a User Story.....	3
Topic: Making SMART Decisions.....	3
SMART on FHIR Standards / Technical Specifications.....	3
Essential Technical Skills.....	4
Infrastructure and Deployment Environments.....	4
Cloud Platforms: AWS, Azure, or GCP (especially managed FHIR services).....	5
DevOps: CI/CD pipelines, containerisation (Docker), monitoring.....	5
Version Control: Git and GitHub for collaboration.....	5
SMART on FHIR: Main Components.....	6
Buy, Build or Adapt?.....	7
Option 1: Buy a commercially available app.....	9
Option 2: Build a new app.....	10
Option 3: Adapt an existing app.....	10
Market Scanning.....	10
Topic: Implementation Planning and Risk Mitigation.....	16
Identifying and Managing Risks.....	19
Managing Project Risk through a Robust Implementation Plan.....	23
Topic: Visualising Your Idea Creating a Lo-Fi Prototype.....	24
Prototypes and Fidelities.....	24
Community Forums and Collaboration Platforms.....	26
Testing, Sandboxes and Tooling.....	28

Topic: Organisational Contexts, Workflows and User Journeys

How to Document Workflows: Key Elements

The below matrix outlines some of the elements that should be documented in a workflow. Each section outlines what to capture, how to document, and provides an example.

Element	What to capture	How to document	Key questions	Example (AU healthcare / SMART on FHIR)
Trigger	The event that starts the work and who/what detects it	One sentence + condition; note source system/signal	What starts this every time? Can it start in more than one way?	'Recall due' when last HbA1c > 6 months (FHIR Observation LOINC 4548-4), flagged by nightly job
Actors / Roles	People and systems involved; permissions and accountability	Role glossary; optional RACI	Who is accountable vs executor? Any external parties?	GP, Practice Nurse, Reception, EHR, FHIR server, Lab, Patient Portal
Steps & Decisions	Ordered tasks; IF/THEN/ELSE branches; error paths	Numbered list or swimlane diagram; 'verb + object' steps	What info moves each step forward? What happens on 'no'?	1) Launch SMART app 2) List abnormal results 3) If HbA1c ≥ threshold → create recall
Hand-offs	Transfers between roles/systems; channel and artefact	Arrows between lanes; include SLA	What makes a hand-off complete? What gets lost today?	GP → Nurse: Recall task with due date; EHR → Pathology: order via secure messaging
Inputs / Outputs	Data and artefacts consumed/produced at each step	Small I/O table per step	Which fields are mandatory? Which codes/units?	Input: Observation history; Output: Task (recall), Communication (patient msg), note
Systems & Data	Applications, APIs, profiles; read/write calls	Context diagram + FHIR resource list/interactions	Where is source of truth? What's read vs written?	FHIR R4 (AU Core): Patient, Observation, DiagnosticReport, Task; SMART scopes: launch/patient, patient/*.read, optional patient/Task.write
Constraints	Policies, privacy, safety rules, SLAs, residency	Short constraints register tied to steps	What must never happen? What approvals are needed?	APPs consent alignment; Australian data residency; WCAG 2.1 AA; critical-results phone call rule
Timing &	Cycle times, queue	KPI table with	When does the	Result available → recall

Metrics	limits, error definitions, done criteria	baseline → target and measurement point	clock start/stop? What counts as done?	created ≤ 1 business day; ≥99% critical results actioned within SLA; failed messages <0.5%
----------------	--	---	--	--

Structure of a User Story

Stories are intentionally short and informal, providing flexibility for technical specialists in how the requirement can be met. They often include acceptance criteria (conditions of satisfaction).

This helps ensure everyone knows exactly what success looks like before development starts.

User stories often follow a simple structure:

- As a [type of user]
- I want [some goal or need]
- So that [reason or benefit].

Topic: Making SMART Decisions

SMART on FHIR Standards / Technical Specifications

You will need to be familiar with the following SMART on FHIR standards/technical specifications.

[CDS Hooks](#)

A standardised method for electronic health records (EHRs) to trigger third-party clinical decision support tools at specific points in a care workflow.

[SMART Sandbox](#)

A de-identified testing environment and a public showcase of SMART apps, helping seed a wider app marketplace later adopted by major EHR vendors.

[SMART Health Cards](#)

A verifiable digital or paper copy of a health record (e.g. vaccinations, test results, scannable QR code etc.)

[SMART Health Links](#)

Patient-centric data sharing standard that allows an individual to share their data with a healthcare provider, usually via QR code.

[SMART Bulk Data/Flat FHIR](#)

An API that enables access to patient-level data across a patient population, which is now part of

HL7 standards and widely used in programs like CMS data sharing.

[SMART Markers](#)

A framework for integrating patient-generated health data. It supports a wide range of data types, including the NIH Patient-Reported Outcomes Measurement Information System ([PROMIS](#)), patient-reported outcomes (PROs), and device-based health tests and measurements.

Essential Technical Skills

Review the following technical skills. Self-check, do you have a working knowledge of following technical skills?

Skill	Description
Web Development Fundamentals (if building the user interface)	JavaScript (React, Angular, Vue)
API Integration	RESTful APIs, JSON, HTTP methods
Authentication and authorisation	OAuth 2.0, OpenID Connect, token management (SMART App Launch and Backend Services)
Security	Transport Layer Security (TLS), data privacy best practices
FHIR Libraries	Use of tools like HAPI FHIR, SMART JS client
Testing and Validation	Tools like Inferno, Touchstone, Postman, and HL7 FHIR Validator CLI. Terminology servers such as OntoServer provide access to clinical terminologies like SNOMED CT via FHIR.

Once skill gaps are clear, you can take steps to address them through training, mentoring, targeted recruitment, or other professional development strategies.

Infrastructure and Deployment Environments

Explore the infrastructure and platforms that will host, deploy, and scale your applications in production environments.

Cloud Platforms: AWS, Azure, or GCP (especially managed FHIR services)

AWS HealthLake: A specialised service designed to organise and normalise healthcare data into the HL7 FHIR format, enabling faster analytics, interoperability, and insights [MediumBlog](#)

- Link: <https://aws.amazon.com/healthlake/>

Azure Health Data Services: A suite of purpose-built technologies for protected health information in the cloud, built on FHIR and DICOM standards, offering managed enterprise-grade services [Microsoft Azure](#)

- Link: <https://azure.microsoft.com/en-us/services/healthcare-apis/>

Google Cloud Healthcare API: A fully managed service that bridges existing healthcare systems and apps, with interfaces supporting FHIR, HL7v2, and DICOM [Cloudtcity, L.L.C.](#)

- Link: <https://cloud.google.com/healthcare-api>

Comparison Article: <https://blog.cloudtcity.com/cloud-based-fhir-services-are-they-right-for-you>

DevOps: CI/CD pipelines, containerisation (Docker), monitoring

- **Docker CI/CD Tutorial:** Docker improves continuous integration and deployment by providing consistent environments throughout the development lifecycle, eliminating platform discrepancies [Octopus Deploy](#)
 - Link: <https://octopus.com/devops/ci-cd/ci-cd-with-docker/>
- **CI/CD with Docker Guide:** <https://spacelift.io/blog/docker-ci-cd>
- **CircleCI Docker Pipeline:** <https://circleci.com/blog/build-cicd-pipelines-using-docker/>
- **Container-Based Workloads:** <https://www.cloud-kinetics.com/blog/enabling-ci-cd-pipeline-for-container-based-workloads/>

Version Control: Git and GitHub for collaboration

- **GitHub Official Documentation:** Git is a distributed version control system that lets developers see the entire timeline of changes and collaborate at any time while maintaining source code integrity [GitHub](#)
Link: <https://docs.github.com/en/get-started/using-git/about-git>
- **Beginner's Guide to Git & GitHub:** <https://medium.com/@aminasaheed223/git-and-github-for-beginners-a-guide-to-version-control-and-collaboration-b9d85b873ad6>
- **Interactive Git Tutorial:** <https://ourcodingclub.github.io/tutorials/git/>

- **Version Control for Collaboration:** <https://mozillascience.github.io/study-group-orientation/3.1-collab-vers-github.html>

SMART on FHIR: Main Components

Let's revisit the core components of a SMART on FHIR app. Understanding these components is essential for successful implementation and will help you identify gaps in your team's knowledge and technical expertise.

Component	Description	Role in App	Example
FHIR Server	A system used to store healthcare data in compliance with the FHIR standard	Stores and serves healthcare data	HAPI FHIR, Azure API for FHIR
SMART App Launch	Implementation guide for launching and authorising access to 3rd party apps	Enables secure app launch (standalone or embedded)	Launching from Electronic Health Record
OAuth 2.0 Authorisation	Authorisation framework	Manages authorisation and token-based access	Access token (for authenticated user) with patient/*.read scope
FHIR Resources	A modular component within FHIR e.g. Patient, Observation	Enables consistent data exchange	MedicationRequest, Condition, Encounter
User Interface (UI)	Front-end interface built with web technologies	Displays data and supports user interaction	React dashboard showing patient vitals
SMART Scopes	Define specific permissions for a client application to access FHIR data, and they can also	Limits what data the app can read or write	launch, openid, patient/*.read

	include EHR context information.		
FHIR Client Library	Toolkit for interacting with FHIR APIs	Simplifies data requests and authentication	SMART JS Client, HAPI FHIR
Security Layer	Encryption and privacy safeguards	Protects data in transit and ensures secure token handling	TLS 1.3, secure token storage

Buy, Build or Adapt?

To make an informed decision about a SMART on FHIR app, you should understand both the organisational needs that the proposed app will address and options (if any) currently available in the market.

You may find it useful to refer to the Australasian Institute of Digital Health (AIDH) resource Assessment Framework for mHealth Apps which serves as a reference tool for app developers working on mobile health apps for Australian release.

Once you have conducted your initial research and analysis, you will need to establish how you will deliver your SMART on FHIR app, starting with the decision to build, buy, or adapt.

- Begin by scanning current apps to see what already exists, then weigh each pathway against your needs, timelines, budget, and in-house capability.
- Your key decision points for whether to build, buy, or adapt an app include strategic alignment, cost, time-to-market, customisation needs, and internal capabilities.
- The following is a breakdown of useful skills and knowledge areas in a decision matrix to assist you when developing SMART on FHIR apps.

Requirement	Build	Buy	Adapt
Strategic Alignment	If it's central to your differentiation.	if it's a commodity or support function.	if an existing solution is close but needs tailoring.
Cost Considerations	Higher initial costs but may reduce licensing fees over time.	Offers lower upfront costs but may include recurring fees and limited flexibility.	May offer lower upfront costs but may include recurring fees and limited flexibility.
Time-to-Market (urgency)	If a speedy market entry is critical.	If you can afford a longer development cycle.	If a partial solution can be quickly customised
Customisation & Control	Gives you complete ownership.	May limit flexibility depending on vendor offerings.	Allows moderate customisation
Internal Capabilities	Requires strong in-house development and DevOps capability.	Shifts responsibility to the vendor but requires vendor management.	May still need integration and customisation skills.
Integration & Compatibility	If integration is complex or proprietary.	If the vendor offers plug-and-play compatibility.	If an existing app needs minor adjustments to fit your ecosystem.

Option 1: Buy a commercially available app

Finding an 'Out-of-the-Box' Solution

You can use the SMART on FHIR App Gallery ('All Apps') to see a wide range of apps aimed at

the North American FHIR ecosystem, including apps for patient engagement, clinical decision support, population health, and more.

Key considerations: Here's some things you might consider when selecting and out of the box solution: EHR support, AU Core/FHIR version alignment, SMART launch support, Security and privacy posture, licensing/total cost, and vendor support.

Out-of-the-box apps for the Australian market (based on AU Core) are limited, and currently include:

Name	Description
My Health Record FHIR Gateway	Allows consumer mobile apps (or app developers) to connect to the My Health Record system via a FHIR gateway, so data can be accessed via FHIR APIs.
Smart Forms (CSIRO / AeHRC)	Renders FHIR Questionnaires (using Structured Data Capture spec + SMART App Launch) to capture standardised clinical assessment data. E.g., the Aboriginal and Torres Strait Islander health check.
Telstra Health's Virtual Care Platform & FHIR-native Health Data Platform	FHIR-based/FHIR native platforms for data exchange and virtual care.

Key Message: If you are attempting to find an app in the Australian market, you will need to ensure that they are based on AU Core or face the prospect of additional costs for configuration for the Australian health ecosystem.

Option 2: Build a new app

If you decide to build a new app, start from SMART on FHIR standards and engage your EHR vendor early to navigate governance and approval processes.

You should also use SMART-enabled sandboxes to test Authorisation and FHIR interfaces, and draw on community resources, open-source examples and JavaScript libraries, to accelerate development.

Build a new app resources

Resources available to you include (but are not limited to):

- The Assessment framework for mHealth apps – www.digitalhealth.gov.au
- FHIR Resources (Implementation Guides) - [The Australian Digital Health Agency](http://www.digitalhealth.gov.au)
- Community support including testing and Implementation resources – Sparked (CSIRO, DHDA, HL7AU & ADHA)
- Connectathons via HL7AU – [HL7 Australia](http://hl7.org.au)
- Open-source code for building questionnaires – Smart Forms Renderer (AEHRC / CSIRO)
- Testing SMART apps in a simulated Australian environment
SMART-EHR-Launcher (AEHRC / Oridashi / CSIRO)

Option 3: Adapt an existing app

Whether (and if) you find an AU Core app that is a close match or a non-Au Core app, you will need to consider reconfiguring or adapting the app to meet your needs.

For open-source apps

- **Consultant-led:** external team delivers end-to-end.
- **Consultant-guided:** experts coach your developers through first implementation.
- **In-house:** your team leads; begin with your EHR vendor on how to embed the app; vendors often provide sandboxes and examples.

For closed source apps

Work with the **original** app developer/owner.

Market Scanning

You will find it useful to conduct a market scan, or review, of existing products, services and vendors.

A market scan helps to map the current landscape, identify viable options, and gather information on risks, interoperability, costs, and timescales.

Here is a simple **10 step** practical way to run a market scan that allows for quick execution and evidence-based decision-making.

Step 1 of 10: Define the decision you need to make

- Name the user groups and the top five functional requirements (user stories) the tool must support.
- You should also identify non-functional (how the app will perform tasks), focusing on attributes like performance, security, privacy, scalability and usability.
- Finally, record project constraints like budget and time.
- List three to five success criteria that will be used to select or reject options.
- Write a one-paragraph problem statement, the outcomes you want, and the date you must decide by.

Step 2 of 10: Set the scope and criteria

Describe the functional and non-functional requirements for your app and prioritise them.

A common approach to prioritising requirements is **MoSCoW**:

Must have

(example: Handle access tokens and API requests securely)

Should have

(example: Write capability to update patient records)

Could have

(example: Offline data caching)

Won't have

(example: Features requiring non-standard FHIR extensions)

Step 3 of 10: Build a search plan

Define the categories you will review (e.g. patient apps, clinician apps, data services, integration components, analytics, and safety tooling).

Plan your sources (e.g. vendor sites, app stores, standards registries, open-source catalogues, academic and industry reports, and reference networks).

Step 4 of 10: Create a structured longlist of options

Important considerations for your planning

Capture each candidate in a consistent table so you can compare them.

You could include some or all of the following fields:

- Product, vendor, company size, funding stage, and years in market
- Primary users and main use cases

- Core capabilities and known limitations
- Interoperability and data, for example:
 - FHIR version, support for AU Core or AU Base, profiles, bulk export, terminology support
 - SMART on FHIR launch (connection and interaction e.g., EHR launch, Standalone launch, Backend Service)
 - SMART scopes required i.e. what an app is allowed to access on behalf of a user or system
- Security (if Cross-origin Resource Sharing – CORS - is required) and privacy posture, certifications, data residency (does the app have data storage and where extracted data is stored), audit features
- Regulatory status where relevant, accessibility conformance, languages
- Deployment options, environments, and typical implementation effort
- Commercials, pricing model, licensing, reference customers and case studies

The following is a **concise comparison** of SMART on FHIR apps that includes a mix of clinical tools, patient apps, and reference implementations) comprised from vendor pages in the [SMART App Gallery](#).

‘FHIR compatibility’ reflects what the gallery page states (e.g., DSTU2, STU3, R4). Some mature apps support multiple versions; verify the exact version with the vendor’s current build. ‘EHR support’ is what the listing claims (often a subset of where the app has been deployed). Many SMART apps are portable but require provisioning per health-system domain (e.g., Epic/Oracle Health processes). (fhir.epic.com, [Oracle Docs](#)).

App	Vendor	Primary Audience	Key use case	FHIR compatibility	EHR Support	License / Pricing	OS
Aidbox Forms	Health Samurai	Patients & clinicians	Launch questionnaires from EHR or patient portal; saves Questionnaire Response	R4	(Gallery lists as SMART app; EHR-agnostic pattern)	Open source	Web (gallery.smarthealthit.org)

Step 5 of 10: Apply ‘knock-out’ screening to create a shortlist

Key Message

Remove entries that fail any ‘must have’ requirements. Keep a note of why each was excluded so you can justify the shortlist.

Step 6 of 10: Evaluate your shortlist

- Try it out. Test it on real tasks by setting up a trial or sandbox and walk through use cases. If this isn't possible, you can create a low-fidelity prototype simulating the user experience and test it out with your users/stakeholders.
- Check that it is fit for purpose. Does it improve workflow? Can data be imported and exported, and can it take your data with you if you leave?
- Assess safety, security and risk. Run a standard security questionnaire. Confirm where it is hosted, who can see logs, and how long data is kept. Identify what could go wrong. If it is clinical, review the vendor's safety case and see how the system fails and recovers. Understand how it is deployed, monitored, backed up and restored, and what support you get (including SLAs)
- Check for ease of use and accessibility. If possible, have representative users complete common tasks and review basic accessibility (WCAG 2.1 AA) so it works for all users.
- Assess costs. This includes costs such as license fee, set-up, training, change management, ongoing costs, and exit costs.

Using a Sandbox

Your set-up and configuration of your sandbox is likely to vary from vendor to vendor. Generally, you will need:

- Docker & Docker Compose (for local sandboxes)
- Java 11+ (if you run Synthea or Java tools)
- curl, Postman (or any REST client), jq (optional for pretty JSON)
- Git if cloning repos (SMART Dev Sandbox, HAPI starter)
- A browser to inspect .../fhir/metadata and CapabilityStatement

Sandbox setup steps

1. Select your sandbox type (e.g. public developer, local or cloud managed).
In this example, we have opted for a public development - using HAPI FHIR.
2. Set up the HAPI FHIR server (you will need to clone the setup from [GitHub](#))
3. Configure for Australian compliance
 - Implement the AU core FHIR Implementation Guide

- Account for My Health Record and Health Identifier Services
 - Ensure privacy and security models Australian requirements
4. Load Australian specific resources
 5. Test your sandbox with FHIR REST APP Calls and
 6. Validate against Australian profiles

Key resources

[HAPI FHIR](#)

[AU Base IG](#)

[ADHA FHIR Resources](#)

[GitHub Starter](#)

Key Message: Never use public health information in a sandbox.

Instead, use synthetic patient generation via open-source sites such as 'Synthea.'

Step 7 of 10: Score and weight the options

Use a simple weighted score so trade-offs are explicit. The exact criterion and weights will depend on your priorities.

An example market score card is provide below with the following weights (adding up to 100): functional fit (30), interoperability (20), security and privacy (20), usability (10), operability (10), commercial cost (10).

Example Market Scan Scorecard

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and data	20	16	12	A supports SMART launch and required

				FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline

Step 8 of 10: Reference checks and risk review

Speak to other users and stakeholders to ask what worked well and what failed in their experience of using the service or product.

Review the contract terms, data processing addendum, service levels, and exit clauses. Ensure risks and their associated mitigations are identified.

Step 9 of 10: Recommendation

Close with a one-page recommendation that states the decision, evidence, costs, and next steps. Tips:

- Use minimal scopes first; add writes only when justified.
- Keep an assumptions/risk log and record evidence (time on task, errors) from quick user tests against the sandbox.
- Load or choose datasets and value sets aligned to AU Core / SNOMED CT-AU / LOINC where possible.

Step 10 of 10: Outputs

The outputs of your market scan should include:

- A longlist with screening outcomes
- A shortlist scorecard with weights and evidence links
- An outline of costs over time
- A risk register and mitigation plan
- A one-page decision paper and timeline

Topic: Implementation Planning and Risk Mitigation

Essential Technical Skills

Plan Components	Question addressed	Brief description
Objectives, scope and success criteria	What problem is the app solving? What is out of scope?	State the pain point, desired outcome, and how you'll measure success.
Stakeholders and Governance	Who are the stakeholders (clinicians, patients, admins)? Who will make decisions and under what authority?	List roles, contexts, and key tasks the app must support.
Readiness assessment	Is your FHIR server conformant (e.g., AU Core)?	Confirm FHIR version, AU Core profiles, and terminology bindings.

	Do you support OAuth 2.0 and SMART App Launch?	Ensure OIDC/OAuth with PKCE, SMART scopes, and discovery endpoints are supported.
Architecture and environments	Standalone or EHR-embedded launch?	Pick the launch pattern that fits workflow and passes patient/encounter context.
	What is the data flow, token handling, and app permissions?	Map sequences, storage/expiry, and least-privilege scopes.
Interoperability factors	What are the factors that enable interoperability?	• Semantic conformance (data standard) • Security and authorisation (access and privacy controls) • Data mapping and quality (migration and transformation logic) • Vendor Consistency and Testing (conformance and validation requirements) • Organisational (workflow, how will the solution support users)
Product and user experience	What problem will the app solve? Will it meet the needs of the end- users?	Personas and journeys: clinician, patient, admin. Minimum viable product (MVP) feature set: the smallest set to deliver value and retire pain points. Accessibility and equity considerations. Content and language guidelines for patient-facing features.
App Development or Integration	How to build using SMART on FHIR standards.	• Implement launch, scopes, and interactions exactly as specified by FHIR standards.
	Does the app succeed in test environments?	• Iterate with synthetic data before touching production. • Utilise sandboxes for testing.
Security, Compliance	Implement TLS 1.2+	• Enforce encryption, short-lived

and Risk Analysis	and secure token management.	tokens and audit trails. Adhere to Australian health standards and regulations (Privacy Act, ADHA standards, NSQDMH standards, State and Territory Health Records Legislation etc.)
	Does it align with Australian privacy laws and ADHA guidelines?	- Informed consent at collection of private data - Ensure Australian data residency, - Follow local policy.
	Have all risks been identified and addressed?	- Risk management and risk register - Threat modelling and mitigations, transport security and headers, storage of secrets
Testing & Validation	Does it pass validation testing?	- Catch profile/API mismatches early; run in CI. - Reference conformance standards. - Resource validation (semantic and syntactic) - Security testing
	Does it meet usability requirements?	- Task-based sessions (usability testing with end users) - Check accessibility (WCAG 2.1 AA).
	What is the product and user experience?	How stakeholders/users perceive and use that service/product across the journey
Deployment & Monitoring	What is the roll out plan?	- Roll out in phases (pilot → full). - Start small, gather evidence, scale with clear exit criteria.
	What monitoring is in place?	- Monitor performance, security, and feedback. - Track logs, metrics, incidents; action on tickets and surveys.
Training, Support and Communications	How will users be onboarded?	Quick reference guides, short videos, role-based training.

		Cutover: go-live date, floor walking, service desk runbooks, hypercare period.
	What are the support channels and documentation?	- Communications: who needs to know what and when. - SLAs, escalation paths, and a living knowledge base.
Evaluation	What, how and when are app metrics evaluated?	- Metrics and baselines: time to complete key tasks, errors, duplicate entry, CDS acceptance, patient waits. - Data collection: logs, analytics, simple samples if needed. - Frequency: weekly during pilot, monthly after go-live.
Costs	What are the costs and compatibility considerations for EHR systems?	- Enterprise EHR System (e.g., Epic, Cerner) - Cost of SMART/FHIR implementation (may be bundled with overall licensing/upgrade fees) - Maintenance/API fees - Training & support

Identifying and Managing Risks

SMART on FHIR is potentially transformative, but it comes with a range of challenges. This section identifies the key risk areas associated with SMART on FHIR apps along with suggested mitigation strategies.

Security and Privacy Risks

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and data	20	16	12	A supports SMART launch and required FHIR profiles

Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline

Interoperability and Legacy System Integration Challenges

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and data	20	16	12	A supports SMART launch and required FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.

Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline

Regulatory and Compliance Challenges

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and data	20	16	12	A supports SMART launch and required FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years

Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline
--------------	------------	-----------	-----------	--

Technical and Resource Limitations

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and data	20	16	12	A supports SMART launch and required FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with

				screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline
--	--	--	--	---

Managing Project Risk through a Robust Implementation Plan

A robust SMART on FHIR implementation plan minimises problems by identifying and addressing risks well before deployment. The table below sets out a baseline set of mitigation strategies to weave into an implementation plan, aligned to the common risk factors identified earlier.

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and data	20	16	12	A supports SMART launch and required FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your

				market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline
--	--	--	--	---

Topic: Visualising Your Idea Creating a Lo-Fi Prototype

Prototypes and Fidelities

Selecting the appropriate level of prototype lets you:

- match effort to risk,
- communicate ideas,
- iterate quickly,
- avoid premature commitment to design concepts.

Prototypes can be categorised into three levels of fidelity: Low, Medium, and High. Your fidelity choice will depend on what you want to test and which stage of the development process you're in.

Comparison table

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.
Interoperability and	20	16	12	A supports SMART

data				launch and required FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline

The most effective approach for your SMART on FHIR app project is to progressively increase in fidelity - starting low and moving higher as your design solidifies.

Key Message: Use Lo-Fi to test concepts quickly, Mid-Fi to validate interactions, and Hi-Fi to demonstrate the final vision.

Low-fidelity prototype formats comparison table

Criterion	Weight	Product A	Product B	Notes
Functional fit to user stories	30	24	18	Covers common user stories.

Interoperability and data	20	16	12	A supports SMART launch and required FHIR profiles
Security and privacy	20	18	14	A has stronger logging.
Usability and accessibility	10	7	9	B slightly easier to use.
Operability and support	10	9	6	A is simpler to implement and run.
Commercial and TCO	10	6	8	B cheaper over three years
Total	100	80	67	The outputs of your market scan should be as follows: - a longlist with screening outcomes - a shortlist scorecard with weights and evidence links - an outline of costs over time - a risk register and mitigation plan - a one-page decision paper and timeline

Community Forums and Collaboration Platforms

Community Forums and Collaboration Platforms support your SMART on FHIR learning journey. Start with introductory resources for newcomers, then progress to the official SMART App Launch specification (STU 2.2) for technical details.

The SMART Health IT hub provides hands-on tutorials, testing sandboxes, and peer discussion forums. Finally, server quick start guides help you implement what you've learned in a working environment.

Global SMART on FHIR Community

- [SMART Health IT Discussion Group](#) - For developers and implementers to share ideas, ask questions, and collaborate.
- [FHIR Community Chat](#) - Zulip-based chat for broader FHIR discussions.

Australian Government Initiatives, Policy and Strategy

- [ADHA Interoperability Plan](#) - This sets out 10 principles to accelerate the shift towards a more interoperable national healthcare system.
- [Digital Health Standards Catalogue](#) - Consolidates all relevant national and international standards into a “single source of truth” for developers, implementers, and users of digital health solutions.

Australian FHIR Community Platforms

- [Sparked](#) – A collaboration between the Department of Health and Aged Care, Australian Digital Health Agency (ADHA), and CSIRO's Australian e-Health Research Centre. Sparked is a vibrant community shaping national FHIR standards through clinical and technical design groups, workshops, and events.
- [ADHA Training & Support](#) - Offers SMART on FHIR training courses and workshops for developers, clinicians, and informaticians. Includes CPD points for CHIA-certified professionals.
- [ADHA Developer Portal \(FHIR resources\)](#) — official guides for My Health Record FHIR Gateway and Provider Connect Australia.

HL7 (Australia) Communities

- [HL7 Australia FHIR Work Group \(Confluence\)](#) — working groups, AU Core TDG, meeting notes and local projects; includes pointers to the Australia Zulip streams.
- [HL7 Australia FHIR Connectathons](#) — regular in-person hack/test events

Australian Implementation Guides

- AU Base & AU Core Implementation Guides — national FHIR building blocks and profiles used widely in AU Connectathons. (hl7.org.au, [FHIR Build](#))

SMART on FHIR Fundamentals

- SMART App Launch specification (current STU 2.2) ([HL7](#), [FHIR Build](#)) - Official

capabilities, conformance and examples.

- A list of SMART App Launch [software libraries](#) - Test environments, vendor sandboxes, sample data.
- [SMART Health IT](#) - Background, tutorials and ecosystem links, includes discussion group
- [Server quick-start \(tutorial\)](#) - Step-by-step for enabling SMART launch.
- [Getting started with Browser-based Apps](#)
- [Getting started for EHRs](#)

Terminology Resources

- [Ontoserver](#) (CSIRO AEHRC) - FHIR-native terminology server; foundation of Australia's National Clinical Terminology Service.
- [HL7 AU Terminology Principles](#) — How AU CodeSystems/ValueSets changes are governed; contact via Zulip
- [National Clinical Terminology Service \(NCTS\)](#) - A government service that manages, develops, and distributes clinical terminologies and tools to support Australia's digital health needs.

Testing, Sandboxes and Tooling

SMART on FHIR communities provide everything you need to design and prove a SMART on FHIR app against international standards before production.

They bring together AU Core–conformant FHIR servers and SMART launch sandboxes, conformance validators, terminology services (SNOMED CT-AU, LOINC, AMT), and safe synthetic data so you can test workflows, security, and interoperability end-to-end.

You will find quick-start environments, Postman/FHIR test collections, continuous-integration checks (e.g., Inferno/Touchstone), accessibility and usability checklists, and examples of 'golden patients' for repeatable scenarios, each tailored to local privacy, identity, and data-residency expectations.

- [Inferno on hl7.org.au](#) - Hosted conformance tests for AU Core with public test kits and release notes.
- [AU Core Inferno Test Kit \(GitHub\)](#) - Run locally or via Docker; links to the public Inferno service.

- SMART launch test host (Sparked FHIR Server) (confluence.hl7.org) - Helper app to test SMART EHR launches against the AU Sparked server.

Vendor Sandboxes

[Allscripts](#)

[athenahealth - athenaOne](#)

[Cerner - Provider and Patient Facing Apps](#)

[Epic Provider Facing Apps](#)

[Epic Patient Facing Apps](#)

[Intersystems](#)

[Meditech](#)

Data Generators

- [Synthea](#) - Open source synthetic FHIR data generator
- [SMART Test Data](#) - 60 de-identified records with Python to generate FHIR from CSV

Sample Apps

- [SMART App Gallery](#) - Listing site with commercial and open-source SMART on FHIR apps)
- [SMART on FHIR DSTU2](#)- JavaScript Example Apps (see also [R3](#))
- [Bulk Data Client - JavaScript](#) - Client-side application or library written in JavaScript that interacts with a bulk data server to retrieve or manage large volumes of data.
- [SMART Growth Chart](#) - Full featured app that has been deployed in care settings. Note that the open-source version of this app requires review before production deployment and is not supported for clinical use.
- [SMART BP Centiles](#)- Full featured app that has been deployed in care settings. Note that the open-source version of this app requires review before production deployment and is not supported for clinical use.
- [Cerner ASCVD Risk Calculator](#)- A software application that helps healthcare providers estimate a patient's risk of developing atherosclerotic cardiovascular disease (ASCVD) over the next 10 years and throughout their lifetime

Design Resources

- [Guide to wireframing \(Figma\)](#)

- [Nielsen-Norman Group](#) – A wealth of resources on a range of user experience topics from two of the pioneers in Human Computer Interaction: Jakob Nielsen and Don Norman.
- [The Interaction Design Foundation](#) - The Interaction Design Foundation is the biggest online design school globally. The site includes a wealth of articles describing design practices and artefacts.