Hardian Health

Innovate UK

Health Innovation Network South London

# Module 8: General XR and health software standards (non-medical device)

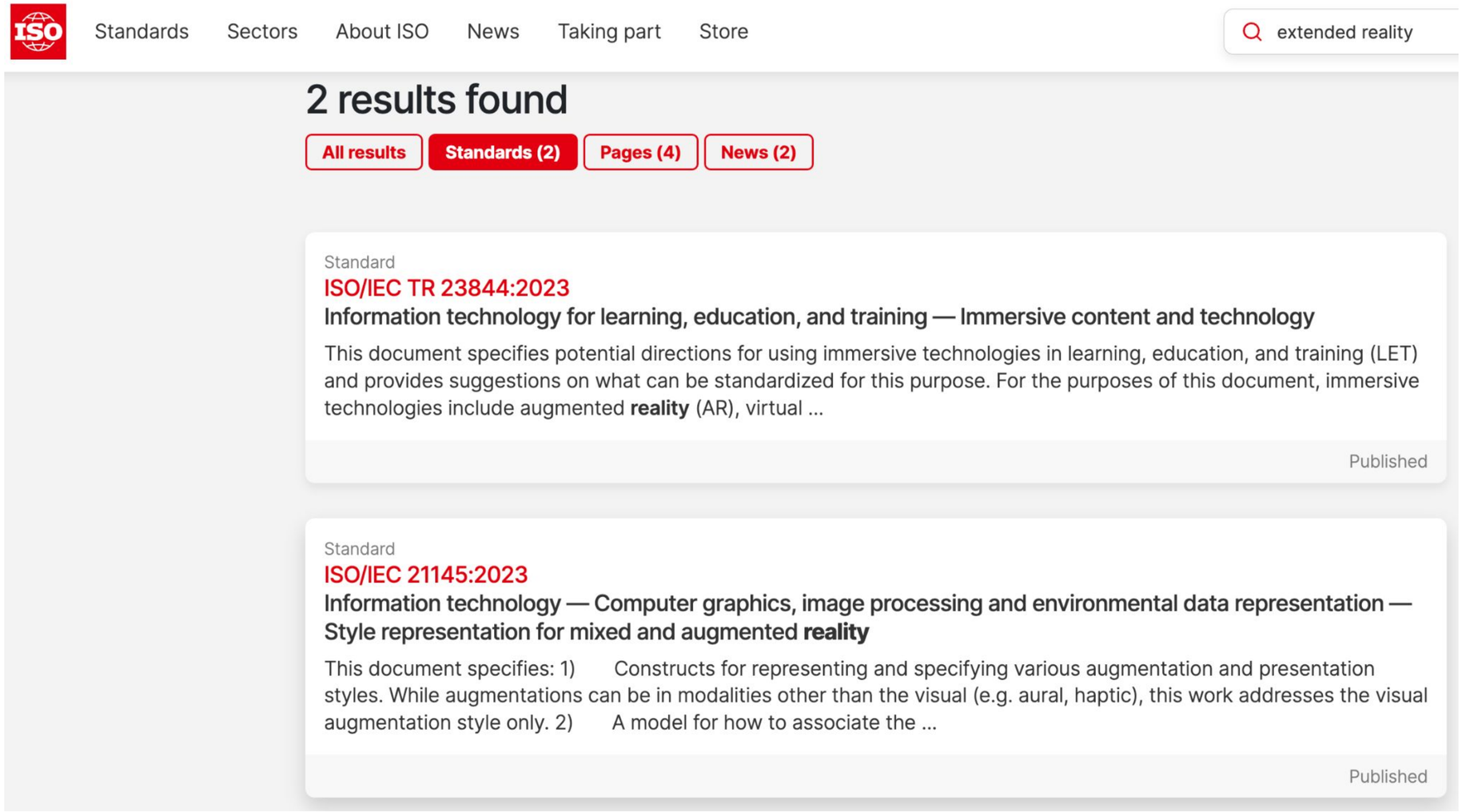
How to apply a standards-based approach to XR development

# Outline

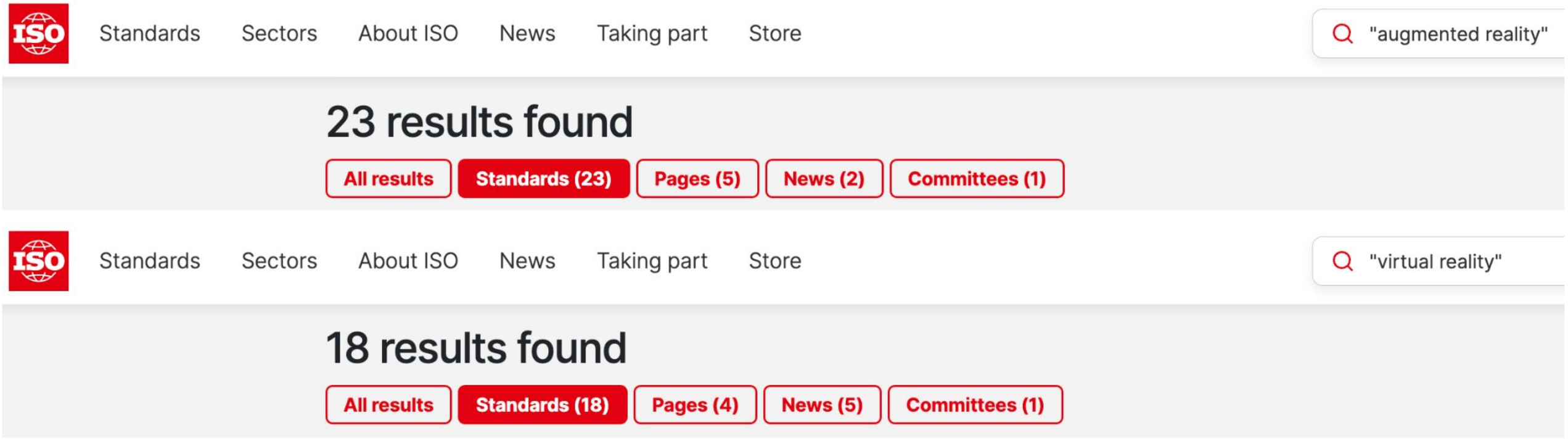
This module will cover:

- XR standards
- Relevant medical device standards - software
- Standards-based software development lifecycle processes
  - Planning and design input: requirements and risk analysis
  - Design output: architectural and detailed design, code and unit test
  - Design verification: software system testing
  - Design validation: usability and clinical evaluation
  - Design review and design transfer: release to market
- Requirements and risk management
- Control of change/ documentation/ identification and traceability
- Multiple choice questions

# XR standards



# XR standards




**For devices that incorporate software or for software that are devices in themselves, the software shall be developed and manufactured in accordance with the state of the art taking into account the principles of development life cycle, risk management, including information security, verification and validation.**

EU MDR GSPR #17.2

# Relevant medical device standards - software

**BS EN 62304:2006+A1:2015** (IEC 62304:2006/Amd 1:2015)

bsi.knowledge

What is BS EN 62304 - *Medical device software* about?

BS EN 62304 deals with medical device software. The amendment adds requirements for legacy software which will help manufacturers demonstrate compliance with the relevant European Directives.

Who is BS EN 62304 - *Medical device software* for?

BS EN 62304 on medical device software is relevant to manufacturers of:

- Medical electrical equipment
- Medical devices incorporating software
- Software that is itself a medical device

# Relevant medical device standards - software

**BS EN 82304-1:2017** (IEC 82304-1:2016)

bsi.knowledge

What is this standard about?

This standard applies to the safety and security of health software products designed to operate on general computing platforms and intended to be placed on the market without dedicated hardware. Its primary focus is on the requirements for manufacturers.

It covers the entire lifecycle including design, development, validation, installation, maintenance and disposal of health software products.

This document aims to provide requirements for the safety and security of health software products; it can only provide such requirements for software-only products. Situations where health software is a part of, or embedded in, a physical device are outside the scope of this document as these combined products are considered separately in, for example, IEC 60601-1 and associated collateral and particular standards.

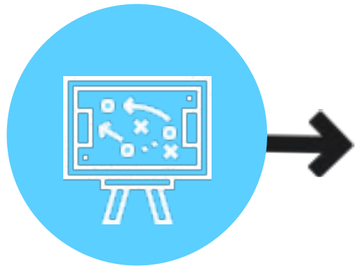
Who is this standard for?

This standard has been written for manufacturers of health software, as it provides them with product requirements for the entire lifecycle.

# Standards-based software development lifecycle processes

## Fundamentals

- Define Intended Use
- Document Regulatory Strategy
- Determine Device Class(es)
- Plan Quality Management System (QMS)
- Plan technical documentation
  - Medical Device File (MDF)
    - *Device Master Record (DMR)*
    - *Design History File (DHF)*
    - *Device History Record (DHR)*



## Development

- Develop product to
  - Clinical Evaluation Plan (CEP)/ Performance Evaluation Plan (PEP)
    - Perform systematic literature review
  - Risk Management Plan (RMP)
  - Software Development Plan (SDP)
  - Cybersecurity Management Plan (CMP)
  - Verification & Validation (V&V) Plans
- Establish QMS

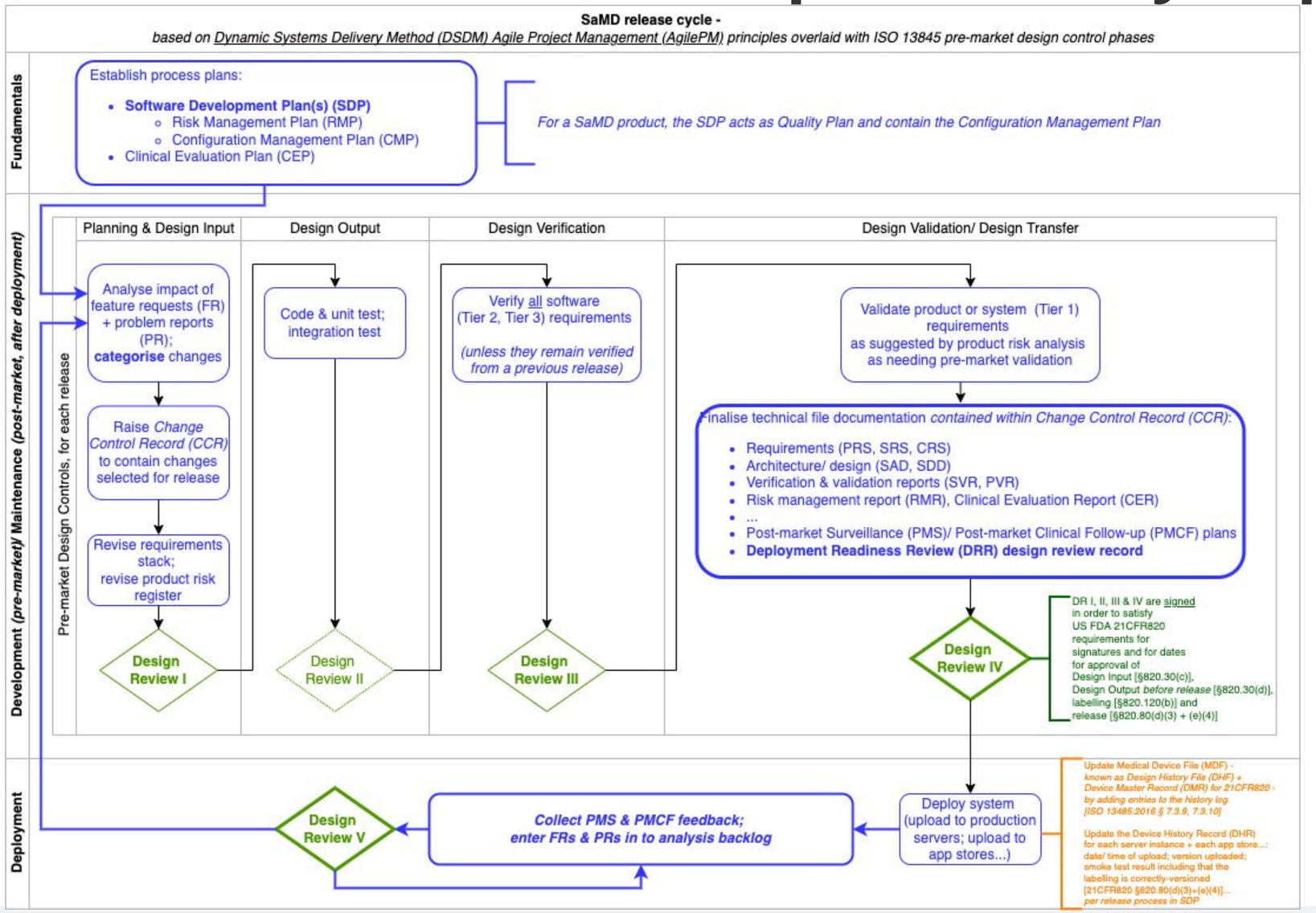
## Deployment

- Deploy product
- Conduct
  - Postmarket Surveillance (PMS)
  - Post Market Clinical Follow-up (PMCF) or Post Market Performance Follow-up (PMPF)

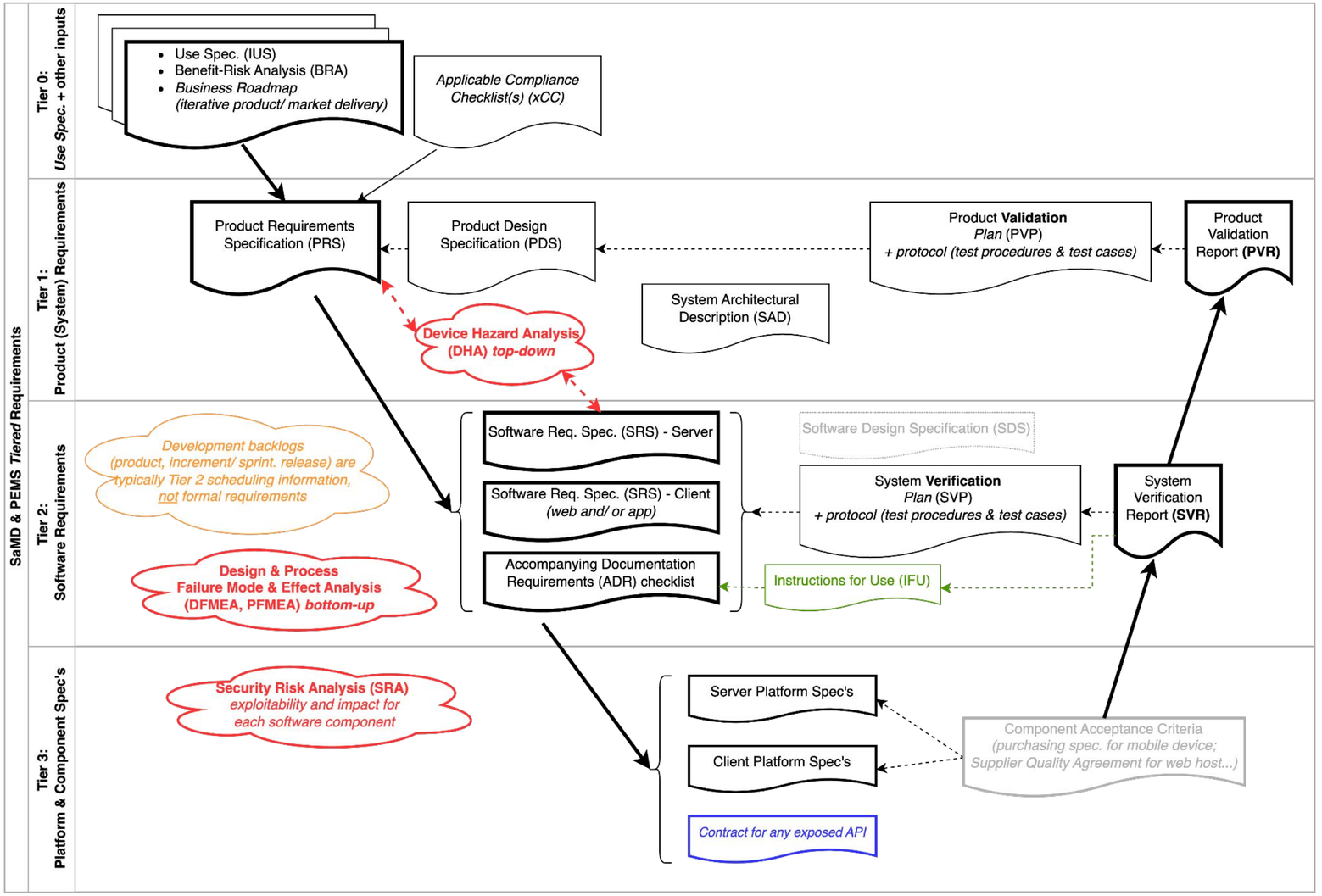
## Maintenance

- Periodically report PMS/ PMCF
- Periodically update Clinical/ Performance Evaluation Report (CER/ PER)/ literature review
- Make *controlled changes* to product from
  - Feature Requests
  - Problem Reports
- Keep on top of cybersecurity: vulnerabilities and possible exploits
- Maintain QMS

# Standards-based software development lifecycle processes



# Requirements and risk management



# Requirements and risk management

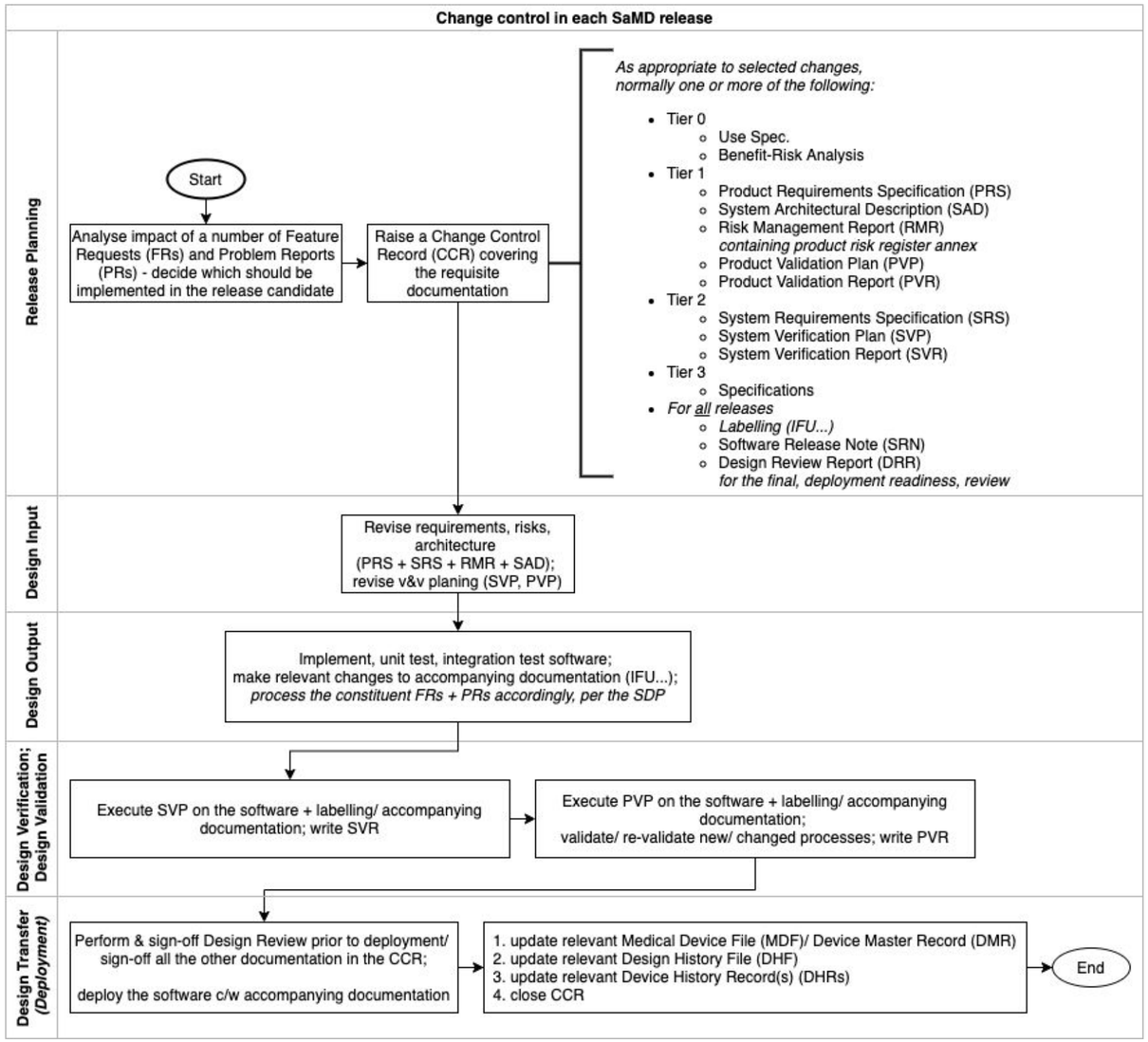
| *Dimension* | HAZARD | Foreseeable sequence of events involving software | Initiating cause | HAZARDOUS SIUTATION | HARM |
|---|---|---|---|---|---|
| *Patient safety* | Electrical energy | Software output controlling laser power applied to an optical coherence tomography (OCT) retinal scan is too high | 1. Software algorithm limitations<br>2. Software is correctly specified but has an ANOMALY | Excessive laser power applied to the patient's eye | • Burns<br>• Loss of vision |
| | Loss of clinical function | Software fails to provide diagnostic function for which it is designed | 1. Software is unable to handle unusual input data<br>2. Software fails to detect incorrect setup of equipment<br>3. Hardware does not provide sufficient resources to support the timely operation of software | 1. Device cannot provide *triage/ detection/ diagnosis* when needed<br>2. Device operates when not correctly setup<br>3. Device fails to warn of injurious or life-threatening condition | • Deterioration of patient's condition<br>• Death |
| | Neglect of patient by clinical staff | Software inputs and outputs confuse of mislead the user<br><br>*(user-related risk analysis aspects of usability engineering)* | 1. Software-human interface confuses the user<br>2. Software outputs exceed the user's capacity to respond<br>3. User does not understand the software's limitations | 1. Mistreatment of patient<br>2. Lack of response to emergencies<br>3. Over-reliance on software replaces personal initiatives *("automation bias")* | |
| *Cyber-security* | Loss of data integrity | *See Table F.1 in PD CEN ISO/TR 24971:2020, the guide to the general application of the medical device risk management standard ISO 14971* | | | |
| | Loss of data availability | | | | |
| | Loss of data confidentiality | | | | |

# Requirements and risk management

| *Dimension* | HAZARD | Foreseeable sequence of events involving software | Initiating cause | HAZARDOUS SIUTATION | HARM |
|---|---|---|---|---|---|
| *AI/ML* | Data quality: incomplete data | *See Table B.3 in BS/AAMI 34971:2023, the guide on the application of medical device risk management to machine learning in artificial intelligence* | | | |
| | Overfitting | | | | |
| | Bias: proxy variables | | | | |
| | Data storage: malicious ML | | | | |
| | Overtrust: overconfidence | | | | |

# Control of change

- **Tier 0 changes** - *significant changes to design and/ or operating principle*
  - *Must be discussed with regulatory authorities (UK Approved Body, EU Notified Body, US FDA)*
  - *May need the product to be recertified*
- **Tier 1 changes**
  - Changes to product (black box) requirements generally need *revalidation*
    - Usability
    - Clinical
- **Tier 2 changes**
  - Changes to system/ software requirements - generally need *reverification*
- **Tier 3 and Tier 4 changes**
  - Minor fixes, or internal changes not visible to users



# Summary

There are standards for XR software that are useful to consider to drive the requirements for your own application; there are also standards that you must implement when realising medical devices including software as a medical device.

The standards help you to implement and coordinate *design control*, *clinical evaluation* and *risk management* to assure safety, effectiveness and cybersecurity of your device.

“The Software Development Life Cycle is for life, not just for Christmas”: that is, you must control change throughout the lifetime of the device.

Contact:

**mike@hardianhealth.com**

**hin.mindset@nhs.net**

Hardian Health

Clinical | Digital | Consulting

www.hardianhealth.com

@HardianHealth

Innovate UK

Health Innovation Network South London