CYBAGE
Delivering Value. Scientifically.

25 Skills a Healthcare Solutions Developer Should Have

Table of Contents

01	**Overview**
02	**Interoperability**
03	**Technical Skills**
04	**Domain Concepts**
05	**Non-Domain Concepts**
05	**Soft Skills**

OVERVIEW

An IT solution professional who has worked on multiple domains including Healthcare will agree to the fact that Healthcare is different for it is driven by regulations (HIPAA, HITECH, and FDA among others). Depending on the product under consideration, the regulations offer multiple competing standards and vocabularies to cater. Subsequently, the business workflows are more complex than others. This raises an interesting question, 'what additional skills does it take for a plain vanilla software developer to be successful in healthcare?' Given the broad and diverse nature of this domain, the nature of solutions, and consequently, the skills required to implement them differ as well. This leads to the next question 'does healthcare IT developers need specific, super specializations for a given solution domain?'

In The Checklist Manifesto, Atul Gawande writes, "We live in the era of the super-specialist—of clinicians who have taken the time to practice, practice, and practice at one narrow thing until they can do it better than anyone else. They have two advantages over ordinary specialists: greater knowledge of the details that matter and a learned ability to handle the complexities of a particular job." He then goes on to cite changing face of operating room, stating it as an example, "We have anesthesiologists just to handle pain control and patient stability, and even they have been divided into subcategories. There are pediatric anesthesiologists, cardiac anesthesiologists,

obstetric anesthesiologists, neurosurgical anesthesiologists, and many others. Likewise, we no longer have just 'operating room nurses.' They too are often subspecialized for specific kinds of cases."

Of course, we do not yet need the level of super-specialization which Atul Gawande is discussing but there is no denial that skills required for one solution domain may not be necessarily relevant to the other. In this whitepaper, we will try to list down the skills of healthcare IT solution developer and then go on to map the relevance of the skills to a given solution domain.

For the sake of brevity, we have restricted the discussion only to care delivery domain and not insurance. Also, we have broadly categorized the solution domains as Population Health Management, Care Delivery Workflows (core and ancillary), Non-clinical Workflows, Patient Engagement, and Integration. Similarly, the skills are categorized as interoperability, domain concepts, non-domain concepts, technology skills, and soft skills. Though these are broad buckets, however, they will offer enough perspective to get the discussion going.

INTEROPERABILITY

1) **Integration standards:** This can be called as the glue of healthcare technology ecosystem. Health Level-7 (HL7 standards is the most popular one. HL7 2.x is the most widely adopted interoperability standard, with HL7-FHIR gaining a lot of traction recently. However, depending on the solution area, the software developer might be required to know other standards as well such as ANSI X12n 5010, NCPDP SCRIPT, DICOM and HL7 v3 CDA.

2) **Semantic vocabularies:** Semantic vocabularies complement the syntactic standards just discussed. Semantic vocabularies ensure that clinical data is interpreted as it should. Some of the widely used semantic vocabularies include ICD-9/10, LOINC, RxNorm, and SNOMED-CT among others.

3) **Integration frameworks:** Integration frameworks constrain the already existing interoperability standards and provide a precise mechanism for syntactic and semantic interoperability, covering an entire clinical workflow. Popular integration frameworks include the now deprecated Healthcare Information Technology Standards Panel (HITSP) standards and widely adopted Integrating the Healthcare Enterprise (IHE). Given that major EHR vendors participated in IHE Connectathon, it makes sense for healthcare developers to get acquainted with the IHE standards.

4) **Integration engines:** These are the workhorses of the interoperability solution arena. Unlike 15 years back, when major products had their own custom built integration engines, now with availability of mature commercial integration engines, seldom will any serious solution developer build it. Some of the popular integration engines include Mirth, Corepoint, Cloverleaf, and Rhapsody among others. Though the engines are similar in functionality, each has its unique learning curve.

TECHNICAL SKILLS

5) **NoSQL:** Interestingly, healthcare was one of the first industries to adopt NoSQL. Major EHR systems including Epic, Cerner, and Meditech are based on MUMPS. Though NoSQL hasn't replaced RDBMS systems yet in other EHRs, it does have increasing applicability in healthcare technology space. It finds particular usage in handling personal care and medical device data; Extract Transform, and Load (ETL) systems; and m-health applications among others.

6) **Big data:** With the advent of Population Health Management, the focus now is on extracting intelligence from the existing healthcare data both structured and unstructured. Large EHR players and insurance organizations are particularly well-suited to understand and implement risk stratification models. For those healthcare developers who intend to play a role in Population Health Management space, it makes sense to invest in big data skills.

7) **ETL Frameworks:** ETL frameworks form the bedrock of any analytics implementation. With fragmented EHR and RCM landscape, the ETL frameworks used in healthcare analytics space are complex in nature. For developers working in healthcare analytics space, knowledge of popular ETL frameworks is a must.

8) **Statistical packages:** Risk stratification and prediction models used in Population Health Management space are pretty complex. While the development of such models will be taken care of by specialized statisticians, its implementation and integration has to be carried out by software developers. To this effect, knowledge of statistical packages such as 'R' and its integration in an analytics framework is important.

9) **Visualization libraries:** Effective data visualization is not just important for population health management and analytics solutions, but also in care delivery workflows and patient engagement solutions. Knowledge of popular data visualization libraries can immensely help healthcare software developers in implementing effective solutions.

10) **Multilingual support:** Patient engagement applications have often to cater to a diverse user base. To this effect, knowledge of implementing multilingual support in an application is helpful.

11) **Data security:** It goes without saying that in healthcare, data security is an absolute must. Hence, it is imperative that a software developer will possess knowledge of strong encryption techniques and libraries and secure coding practices.

12) **Third-party integrations or mashups:** All players in healthcare solution space, including big EHR vendors have begun to realize that no single player can offer solutions to the entire care delivery workflow. EHR vendors are now actively promoting smaller niche players to partner and use their systems as a solution platform. All major players including Athenahealth, Allscripts, and Practice Fusion among others have begun to offer what they call marketplaces. Moreover, this 'mashup' culture is here to stay. Hence, it becomes advantageous for a healthcare software developer to learn integration

standards, such as HL7 FHIR, SOAP and REST; and authentication and authorization frameworks, such as OAuth.

13) **Application conformance:** Continuing the discussion on technical solutions and standards governing integration and interoperability healthcare, maintaining conformance holds a key to seamless interoperability in a workflow. Software developers should invest in skills required to design and define conformance and maintain it.

14) **Performance optimization:** Though a typical healthcare application does not deal with millions of transactions across geographical boundaries per second, given the implications, performance is a critical aspect for the success of any healthcare technology solution. It makes sense for healthcare software developers to acquire skills to design for scalability and performance, measure performance, and implement performance optimization techniques through good coding and design practices.

DOMAIN CONCEPTS

15) **Health Insurance Portability and Accountability Act (HIPAA):** This is the single most regulatory aspect, which a healthcare software developer must know and understand. This will help the developer, not just to implement solutions compliant with regulatory framework, but also to follow secure development practices.

16) **Food and Drug Administration (FDA):** Though not as prevalent as HIPAA, FDA regulations are finding increasing applicability in core care delivery space. IT solutions in lap and radiology space usually come along with some kind of FDA approval. Also, software developers working in m-health space need to understand when does that application falls under FDA ambit and what processes one must follow. That said, IT projects involving FDA approval have a process champion with knowledge of FDA nuances.

17) **Care delivery workflows:** A healthcare solution developer must understand what a typical care delivery workflow looks like. This includes understanding concepts such as Encounter, discharge process, care coordination, and specialty taxonomy among others.

18) **Revenue cycle workflows:** Similarly, a healthcare solution must understand how a typical revenue cycle workflow operates. This includes understanding concepts such as billing schedules, super bill, claim fields, payment models, and denial workflows among others. The degree of knowledge of revenue cycle workflow required depends on the working area of the software developer.

19) **Quality frameworks and metrics:** With the focus on delivering quality healthcare and the growing ACO movement, it makes sense for a healthcare software developer to understand how 'quality' is measured in healthcare. The best way to do is to gain expertise over existing quality measurement and improvement frameworks, for example, AHRQ, NCQA, and CMS 33 among others.

20) **Other standards and certifications:** Often healthcare technology solutions need to conform to functional completeness standards. These are either mandated by regulatory authorities such as ARRA HITECH, meaningful use standards or can be by industry standard bodies such as now-defunct CCHIT. Depending on the context, healthcare software developers should know and understand the applicable standards.

NON-DOMAIN CONCEPTS

21) **Statistics:** Lately, there has been a tremendous focus on analytics in healthcare. As noted earlier, while the statistical models might be developed by specialized statisticians. It has to be ultimately implemented by a healthcare solution developer. It makes sense for healthcare software developers to invest in gaining basic to intermediate statistics knowledge.

22) **Usability:** Though usability aspects in larger projects are often crystallized by usability specialists and not software developers, it still makes sense for healthcare solution developers to gain knowledge about aspects of usability and it's applicability in healthcare solutions. Particularly for the developers who are working in startups or smaller projects. It also helps developers to interact with usability experts and provide them inputs for improving the workflow.

SOFT SKILLS

23) **Planning and coordination:** As noted earlier in this discussion, healthcare is all about interoperability. A typical interoperability project involves multiple stakeholders and a lot of moving parts. During an Interoperability Implementation phase, a software developer might be in the thick of things. In the absence of a dedicated project manager, a software developer will have the necessary skills to identify dependencies, formulate a workable plan, and coordinate with other stakeholders.

24) **Interpersonal:** This is kind of corollary to the previous discussion. Since, a software developer might be involved in planning and coordination, decent interpersonal skills will help to pull that smoothly.

25) **Patience:** In the fast-moving world, this might seem counter intuitive. However, large enterprises are slow and healthcare enterprises are even more. While the product may be ready, the implementation and integration aspects might not proceed as smoothly or swiftly as expected. There could be obstacles ranging from availability of personnel, infrastructure, user training, or a non-cooperating third-party vendor. In such scenarios, patience and tact is a must-have skill.

Now that we have listed down the most likely required skills, let's try to map their applicability and relevance to a given healthcare sub-domain.

	Population Health Management	Care Delivery Workflows (core + ancillary)	Non-Clinical Workflows	Patient Engagement	Integration
Interoperability					
HL7 v2.x	Working knowledge	Working knowledge	Working knowledge	Working knowledge	Expert
HL7 CDA (CCD)	Working knowledge	Expert	Awareness	Working knowledge	Expert
HL7 V3	Awareness	Awareness	Awareness	Awareness	Working knowledge
HL7 FHIR	Awareness	Working knowledge	Working knowledge	Working knowledge	Expert
ANSI X12n	Working knowledge	Awareness	Working knowledge	Awareness	Expert
NCPDP SCRIPT	Awareness	Awareness	Awareness	Awareness	Expert
ICD 9/10	Working knowledge	Working knowledge	Working knowledge	Working knowledge	Working knowledge
RxNorm	Working knowledge	Working knowledge	Awareness	Working knowledge	Working knowledge
SNOMED-CT	Working knowledge	Working knowledge	Awareness	Working knowledge	Working knowledge
IHE	Awareness	Awareness	Awareness	Awareness	Expert
Integration engines	Not required	Awareness	Awareness	Awareness	Expert
Technology					
NoSQL	Expert	Working knowledge	Working knowledge	Working knowledge	Working knowledge
Big data	Expert	Awareness	Awareness	Awareness	Not required
ETL frameworks	Expert	Not required	Not required	Not required	Not required
Statistical packages	Working knowledge	Not required	Not required	Not required	Not required
Visualization libraries	Expert	Working knowledge	Working knowledge	Working knowledge	Not required
Multilingual support	Not required	Working knowledge	Awareness	Working knowledge	Not required
Data security	Working knowledge	Working knowledge	Working knowledge	Working knowledge	Working knowledge
Third-party integrations	Awareness	Working knowledge	Working knowledge	Working knowledge	Working knowledge
Application conformance	Awareness	Working knowledge	Working knowledge	Awareness	Expert
Performance optimization	Working knowledge	Working knowledge	Working knowledge	Working knowledge	Working knowledge
Domain Concepts					
HIPAA regulations	Working knowledge	Working knowledge	Working knowledge	Working knowledge	Working knowledge
FDA	Not required	Working knowledge	Not required	Working knowledge	Not required
Care Delivery workflow	Working knowledge	Expert	Working knowledge	Working knowledge	Awareness
Revenue Cycle workflows	Working knowledge	Awareness	Expert	Working knowledge	Awareness
Quality frameworks	Expert	Working knowledge	Working knowledge	Working knowledge	Not required
Other Standards or certifications	Awareness	Working knowledge	Working knowledge	Awareness	Awareness
Non-domain Concepts					
Usability	Awareness	Working knowledge	Working knowledge	Working knowledge	Not required
Statistics	Working knowledge	Awareness	Awareness	Awareness	Not required

This table could also be used as a checklist to pick the right team or help formulate a skill development plan for existing team.

CYBAGE
Delivering Value. Scientifically.

Cybage Software Pvt. Ltd.

HQ: Cybage Towers, Survey No 13A/ 1+2+3/1, Vadgaon Sheri, Pune 411014 |
Tel: 91 20 6604 4700 | Fax: 91 20 6604 1701
Pune | Hyderabad | Gandhinagar | Redmond | New Jersey | California | Atlanta |
Texas | London | Frankfurt | Amsterdam | Sydney | Singapore | Japan
www.cybage.com