HEALTHCARE TECHNOLOGY AND ITS CHALLENGES

By Shameem C Hameed

Hello, welcome to health IT simplified. I'm Shameem C. Hameed, founder of blueEHR and HITaaS, Health IT as a Service. I've spent the last 32 years in digital health and the last 15, in making it affordable and accessible. So, I have some thoughts on the state of digital health today. As you all know the HIMSS conference was concluded recently. They discussed new and fancy technologies that are going to change our lives, like artificial intelligence, big data block chain, Internet of medical things, patient engagement and so on. Now I understand that these are all really good developments that'll improve the delivery of care. However, how do you build new and advanced technologies on an antiquated core? Most of these technologies are being developed on the cloud and are meant to be part of an interconnected seamless eco-system.

So, what do I mean by the antiquated core?

All the new technologies that are being bragged about requires a quarterback if you will. And that is the ERP systems used by healthcare organizations for its operations. These are the systems that are being used to organize and facilitate operations and more importantly for clinical documentation and workflows. So, these systems are where the rubber meets the road. If you want to do big data or artificial intelligence, the data must come from these systems. If you want to do medical devices integrations of internet of medical things, it must be integrated with these EHR systems. If you want to do better patient engagements, it has to flow from these systems. So, you see the entirety of digital innovations depends on the connectivity with these core systems and their ability to connect to these systems.

The problem is that the leading systems are technologically antiquated. The primary 2 of them are based on 1960's technology called M. U. M. P. S or Massachusetts General Hospital utility multi programming systems. Short for M. U. M. P. S. Others use at least a couple of decades old technologies like client server technologies or the like. They are antiquated in code and architectural substructures, impossibly difficult to scale without massive investments, inflexible in adapting to new workflows in both clinical and operational areas, siloed and walled off and most of all not interoperable. The major drawbacks of these EHR systems are **Usability, Scalability, and interoperability**.

Let us talk about usability first. So, we have seen thousands of studies about physician fatigue and the burn out that occurs when they use EHR systems. We have seen people trying to build better user interfaces to rectify these issues. In other words, they're trying to customize these interfaces of these older systems. The customizability of these old system is nearly impossible because of the old technology that it is built on. The systems architectures have changed over time, and we have seen the evolution of applications from

monolithic code to the different frameworks in use today including MVC, the micro-services of a service-oriented architectures. To put very simply, these new architectures segregate different aspects and functions of the application. So that resources and customizations can be allocated to specific areas without affecting each of them. How does that help? It helps in 2 main ways:

1. The interfaces can be changed with better user experience concepts without affecting the underlying database structure or business logic.

2. Flexible adaptations or customizations can be built on the system without much expense. Customizing old systems to make better user experience in the old technology framework can be prohibitively expensive.

That leads to our second point **scalability**. So, continuing on the earlier example of expensive customization, let us assume that we invest in customizing a better user experience for these systems with all the expenses. Those changes are restricted to a particular installation of that particular system. So, if you have multiple installations updating and synchronizing these changes between these systems become very difficult and expensive.

Secondly, scalability is difficult because of the monolithic code structure that the systems employ. Monolithic systems are resource intensive. So let us say that one part of the application alone is being used at a particular time intensively like reporting on multi-year data. There is no way in an older system to allocate or distribute specific resources to that particular area. So, if somebody is generating a big report the entire system will start hanging. That is to put it in a very simple way. So that is where the cloud architecture or being a cloud-native architecture becomes critical. Cloud enables you to scale your resources either vertically or horizontally. It means that you can enhance the server processing capacity of one server, or you can add more servers on the go. So, cloud-native architectures are also service oriented architecture in using micro-services and APIS. Which means that if one micro-service requires more resources, additional resources can be channeled to that area. Again, this means that the rest of the system doesn't hang and that area alone can be given more resources. So, a cloud-based architecture enables your organization to expand with minimal investment and of course it allows access anywhere anytime. And it also outsources most of the basic maintenance and security functions to the cloud provider. Which means that your internal cost of running that system is marginal. So, the current trend among these older systems is to move their software to a cloud and call it as a cloud-based system. We must differentiate between a **cloud hosted system** and **a cloud-based system**.

A cloud hosted system merely moves the onsite hardware into a virtual hardware. Which means that instead of having the server on your local location it is moved to a cloud. It does not exploit the capabilities of a true cloud born and bred system.

And finally let us talk about interoperability. Interoperability is key. The future of healthcare operates in the digital ecosystem. Emphasis is on the word ECOSYSTEMS. Eco system means interdependent systems that should be able to communicate with each other. So, the financial markets and the financial systems have moved on to these kind of technologies long ago. All the new technologies in healthcare spectrum that are being proposed are developed are depended on such kind of communication. They use web communication protocols like APIs and micro-services that are open and easily accessible. And if that communication is not enabled, then the whole digital eco system concept will struggle. So many of these old technology systems do not share data easily for 2 main reasons.

1. They do not want to share their data so that they can perpetuate their monopoly in the market.

2. These systems are not set up with the technological requirements of interoperable. If they are not technologically set up for a true cloud-based ecosystem, how can they become the anchor or the core of the new digital healthcare future?

So dear friends, in the upcoming podcasts it will be my endeavor to try and talk to you about my take on the problems with the healthcare technology today. I will also bring in people who are dealing with the ground level realities of the disconnect between what the technologists deem is required in the market and what the practitioners really want. So, stay tuned and thank you.