International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

From Code to Care: The Impact of Software Engineering and QA on Healthcare Delivery Systems

Nikhil Bharadwaj Ramashastri

Turo Inc, USA

Abstract
The rapid digitalization of healthcare has led to an unprecedented reliance on software systems to improve patient care, streamline operations, and ensure regulatory compliance. This article presents a comprehensive analysis of the critical role that software engineering and quality assurance (QA) play in three key areas of modern healthcare: Electronic Health Records (EHRs), telemedicine platforms, and medical device software. Through an examination of current practices, challenges, and case studies, we demonstrate how robust software engineering methodologies and rigorous QA processes contribute to the development of reliable, secure, and efficient healthcare applications. Our findings highlight the importance of data integrity, interoperability, user experience, and regulatory compliance in healthcare software development. Furthermore, we explore emerging trends such as artificial intelligence, blockchain, and the Internet of Medical Things (IoMT), discussing their implications for future software engineering and QA practices in healthcare. This article provides valuable insights for healthcare organizations, software developers, and policymakers, offering recommendations to enhance the quality and safety of software-driven healthcare solutions.

Keywords: Healthcare Software Engineering, Medical Quality Assurance, Electronic Health, Records (EHRs), Telemedicine Platforms, Medical Device Software.

1. Introduction

The healthcare industry is undergoing a profound digital transformation, with health information technology (HIT) becoming increasingly integral to patient care, operational efficiency, and regulatory compliance. This shift has placed unprecedented demands on software engineering and quality assurance (QA) practices within the healthcare sector. As Sittig and Singh point out, healthcare systems are complex adaptive systems, where the implementation of HIT can have far-reaching and often unforeseen consequences [1]. This complexity underscores the critical need for robust software engineering and QA processes in healthcare applications. From Electronic Health Records (EHRs) that manage vast amounts of sensitive patient data to clinical decision support systems that enable personalized medicine, the stakes for quality and safety in healthcare software have never been higher [2]. As medical software systems grow in complexity and criticality, ensuring their reliability, security, and effectiveness has become paramount. This article examines the crucial role of software engineering and QA in three key areas of modern healthcare: EHRs, telemedicine platforms, and medical device software. By analyzing current practices, challenges, and emerging trends, we aim to provide a comprehensive overview of how robust software engineering methodologies and rigorous QA processes contribute to the development of healthcare applications that not only meet regulatory standards but also enhance patient outcomes and safety in the context of complex, adaptive healthcare systems.

2. Background and Literature Review

The landscape of healthcare software has undergone significant transformation in recent years, driven by rapid technological advancements and evolving regulatory frameworks. At the forefront of these changes is the U.S. Food and Drug Administration's (FDA) Digital Health Software Precertification (Pre-Cert) Program, which represents a paradigm shift in how healthcare software is regulated and quality-assured [3].

2.1. Evolution of healthcare software systems

The evolution of healthcare software systems has been marked by a transition from traditional, siloed applications to more integrated, patient-centric platforms. This shift is reflected in the FDA's Pre-Cert Program, which acknowledges the unique characteristics of software as a medical device (SaMD) and software in a medical device (SiMD).

The Pre-Cert Program recognizes that traditional regulatory approaches, designed for hardware-based medical devices, are not always suitable for the rapid iteration and continuous improvement cycles typical of software development. This recognition has led to a more flexible, risk-based approach to software regulation, focusing on the developer's ability to create high-quality software products rather than on the products themselves.

2.2. Regulatory landscape in healthcare software development

The FDA's Pre-Cert Program represents a significant evolution in the regulatory landscape for healthcare software development. The program aims to provide more streamlined and efficient regulatory oversight of software-based medical devices while still ensuring patient safety and product quality.

Key aspects of the Pre-Cert Program include:

1. **Organization-level certification:** The FDA evaluates the software developer's culture of quality and organizational excellence.
2. **Streamlined premarket review:** Pre Certified companies may be able to market lower-risk devices without additional FDA review or with a streamlined premarket review process.

3. **Real-world performance monitoring:** Ongoing monitoring of real-world performance data to verify the continued safety and effectiveness of the software.

This approach represents a shift from the traditional product-centric regulatory model to one that emphasizes the importance of robust software engineering and quality assurance processes throughout the entire software lifecycle.

2.3. Current challenges in healthcare software engineering and QA

The Pre-Cert Program highlights several key challenges in healthcare software engineering and QA:

1. **Rapid innovation cycles**: The program acknowledges the need for regulatory processes that can keep pace with the rapid development and iteration cycles typical of software.
2. **Continuous monitoring and improvement:** The emphasis on real-world performance monitoring underscores the need for robust post-market surveillance and continuous improvement processes.
3. **Organizational excellence:** The focus on organization-level certification highlights the importance of cultivating a culture of quality and excellence in software development practices.
4. **Risk management:** The risk-based approach of the Pre-Cert Program emphasizes the need for effective risk management strategies in healthcare software development.
5. **Interoperability and data management:** While not explicitly part of the Pre-Cert Program, the increasing integration of healthcare systems underscores the importance of interoperability and effective data management practices.
6. **Security and privacy:** The handling of sensitive health data necessitates robust security and privacy measures, a concern that extends beyond regulatory compliance to ethical obligations and user trust.
7. **User-centered design:** The program's focus on real-world performance implicitly emphasizes the importance of user-centered design in creating effective and safe healthcare software.

These challenges underscore the need for a holistic approach to healthcare software engineering and QA, one that goes beyond mere regulatory compliance to embrace a culture of quality, safety, and continuous improvement.

As the healthcare industry continues to digitize and software becomes increasingly central to patient care, addressing these challenges will be crucial. The FDA's Pre-Cert Program provides a framework for thinking about these issues, but it will be up to healthcare software developers to implement robust engineering and QA practices that can meet these challenges head-on.

3. Electronic Health Records (EHRs)

3.1. Overview of EHR systems

Electronic Health Record (EHR) systems are comprehensive, digital versions of patients' paper charts. They are real-time, patient-centered records that make information available instantly and securely to authorized users. EHR systems are designed to go beyond standard clinical data collected in a provider's office and can include a broader view of a patient's care [4].

Key features of modern EHR systems include:

- Patient demographics and medical history
- Medication and allergy lists
- Clinical notes and discharge summaries
- Laboratory and imaging results
- Billing information
- Decision support tools

The adoption of EHRs has been driven by the need to improve healthcare quality, patient safety, and operational efficiency. In the United States, the Health Information Technology for Economic and Clinical Health (HITECH) Act of 2009 significantly accelerated EHR adoption through financial incentives and penalties [5].

3.2. Software engineering considerations for EHR development

3.2.1. Data model design

The data model is the foundation of an EHR system. It must be flexible enough to accommodate diverse types of medical data while maintaining structure for efficient querying and analysis. Considerations include:

- Use of standardized medical terminologies (e.g., SNOMED CT, LOINC)
- Support for structured and unstructured data
- Temporal data representation
- Patient identification and record linkage

3.2.2. Interoperability standards

Interoperability is crucial for EHRs to exchange data with other healthcare systems. Key standards include:

- HL7 FHIR (Fast Healthcare Interoperability Resources)
- HL7 v2 and v3
- DICOM for medical imaging
- IHE (Integrating the Healthcare Enterprise) profiles

Implementing these standards ensures that EHRs can communicate effectively with other systems, facilitating continuity of care and data aggregation for research and public health purposes.

3.2.3. User interface design for healthcare professionals

EHR user interfaces must balance efficiency, accuracy, and ease of use. Considerations include:

- Workflow-oriented design
- Customizable views for different specialties
- Integration of decision support tools
- Mobile-friendly interfaces for bedside use
- Data visualization for trend analysis

User-centered design processes, involving iterative feedback from healthcare professionals, are essential to create interfaces that enhance rather than hinder clinical workflows.

3.3. Quality assurance processes for EHRs

3.3.1. Data integrity testing

Ensuring the accuracy and consistency of data within an EHR is paramount. QA processes include:

- Validation of data entry and import processes
- Verification of data transformations and calculations
- Testing of data persistence and retrieval
- Audit trail verification

3.3.2. Security and privacy compliance testing

EHRs contain sensitive personal health information and must comply with regulations such as HIPAA in the United States. Security and privacy testing includes:

- Access control and authentication testing

- Encryption verification for data at rest and in transit
- Audit logging and monitoring
- Penetration testing and vulnerability assessments

3.3.3. Performance and scalability testing

EHRs must perform efficiently under various load conditions. Performance and scalability testing involves:

- Load testing to simulate multiple concurrent users
- Stress testing to identify system breaking points
- Scalability testing to ensure performance as data volume grows
- Response time testing for critical operations

These tests ensure that the EHR can handle the demands of a busy healthcare environment without compromising user experience or patient care.

4. Telemedicine Platforms

4.1. The rise of telemedicine and its impact on healthcare delivery

The COVID-19 pandemic has catalyzed an unprecedented surge in telemedicine adoption, transforming healthcare delivery on a global scale. This rapid shift has highlighted both the potential and the challenges of virtual care platforms [6]. Telemedicine has emerged as a critical tool for maintaining healthcare services while minimizing the risk of virus transmission, demonstrating its value beyond the pandemic context [7].

Key impacts of telemedicine on healthcare delivery include:

- Ensuring continuity of care during public health crises
- Expanding access to healthcare services, particularly in rural or underserved areas
- Reducing unnecessary emergency department visits
- Facilitating chronic disease management
- Improving infectious disease control through remote triage and monitoring
- Enhancing healthcare system capacity and efficiency

4.2. Software engineering challenges in telemedicine applications

4.2.1. Real-time communication systems

The core of telemedicine platforms is reliable, high-quality audio and video communication. Engineering challenges include:

- Implementing low-latency, high-fidelity audio and video transmission
- Ensuring compatibility with various devices and network conditions
- Developing secure, HIPAA-compliant communication protocols
- Creating backup communication channels for system resilience

4.2.2. Cross-platform compatibility

As highlighted by Wosik., the rapid adoption of telemedicine necessitated the use of various devices and platforms [6]. Key considerations include:

- Developing responsive interfaces for different screen sizes and resolutions
- Ensuring consistent functionality across web, mobile, and desktop platforms
- Optimizing performance for low-bandwidth scenarios
- Supporting a wide range of browsers and operating systems

4.2.3. Integration with existing healthcare systems

Telemedicine platforms must seamlessly integrate with existing healthcare IT infrastructure. Challenges include:

- Interfacing with Electronic Health Record (EHR) systems
- Implementing standardized data exchange formats (e.g., HL7 FHIR)
- Ensuring secure data transfer between systems
- Managing user authentication and access control across integrated platforms

4.3. QA strategies for telemedicine software

4.3.1. Network resilience testing

Given the critical nature of telemedicine consultations, especially in emergency situations [7], network resilience is paramount. QA strategies include:

- Simulating various network conditions (e.g., low bandwidth, high latency)
- Testing automatic reconnection and session recovery mechanisms
- Verifying graceful degradation of service quality under poor network conditions
- Stress testing servers with high concurrent user loads

4.3.2. User experience testing for patients and providers

The rapid transition to telehealth highlighted the importance of user-friendly interfaces for both patients and healthcare providers [6]. UX testing strategies include:

- Conducting usability studies with diverse user groups, including those with limited technical proficiency
- Testing accessibility features for users with disabilities
- Evaluating the intuitiveness of the interface for first-time users
- Assessing the efficiency of common workflows for healthcare providers

4.3.3. Compliance with telehealth regulations

As telemedicine adoption accelerates, ensuring compliance with evolving regulations is crucial. QA processes for compliance include:

- Verifying adherence to privacy and security regulations (e.g., HIPAA in the US)
- Testing compliance with region-specific telehealth laws and regulations
- Ensuring proper implementation of consent processes and documentation
- Validating the security of data transmission and storage

The COVID-19 pandemic has demonstrated the critical role of telemedicine in global health emergencies and routine care alike [7]. As these platforms continue to evolve, incorporating features like remote patient monitoring and AI-assisted diagnostics, the complexity of development and testing will increase. Robust software engineering practices and comprehensive QA strategies are essential to realize the full potential of telemedicine in enhancing healthcare accessibility, efficiency, and outcomes.

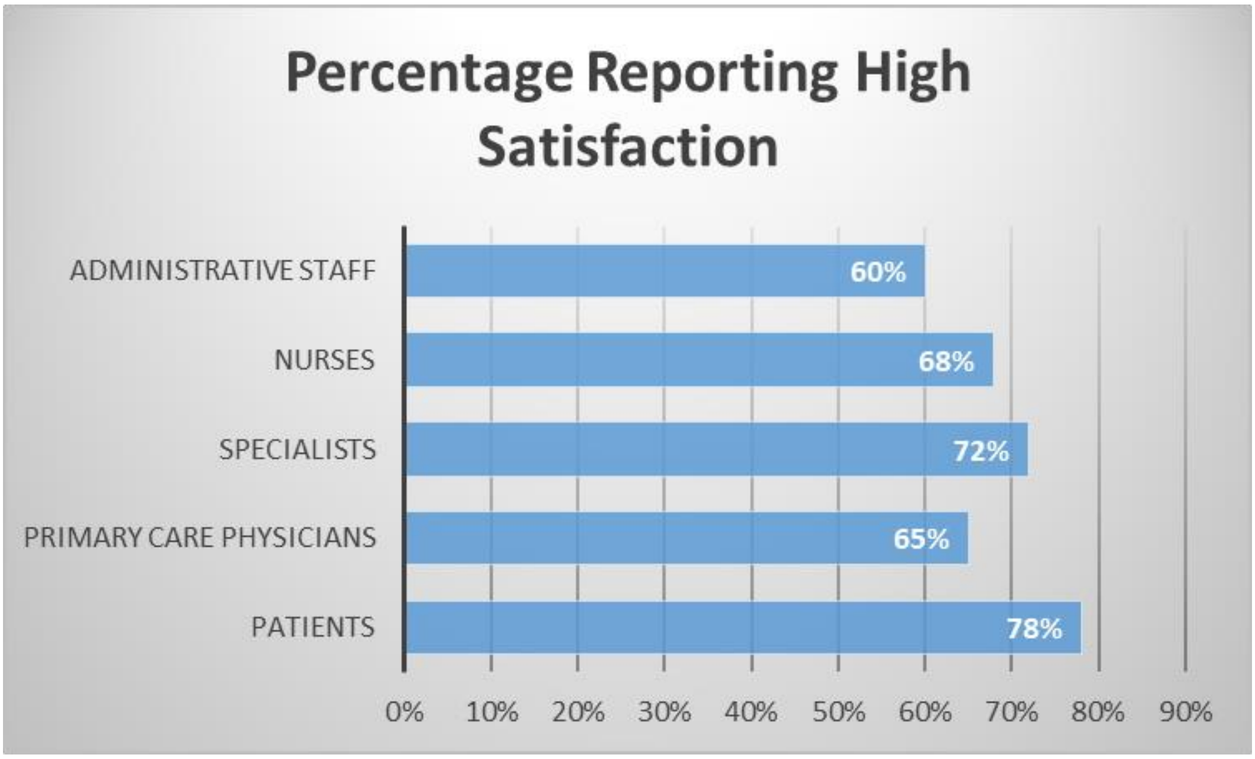

Fig. 1: User Satisfaction with Telemedicine Platforms [10]

5. Medical Device Software

5.1. Types of software-driven medical devices

The FDA classifies medical devices, including software-driven devices, into three classes based on their risk level and the degree of regulatory control necessary to ensure safety and effectiveness [8]:

- Class I: Low-risk devices (e.g., digital thermometers)
- Class II: Moderate-risk devices (e.g., infusion pumps, surgical robots)
- Class III: High-risk devices (e.g., implantable pacemakers)

Software can be a medical device itself (Software as a Medical Device, or SaMD) or an integral part of a medical device. The classification determines the level of regulatory scrutiny and the requirements for premarket submission.

Class	Risk Level	Examples	Regulatory Controls
I	Low	Digital thermometers	General controls
II	Moderate	Infusion pumps, surgical robots	General controls and special controls
III	High	Implantable pacemakers	General controls, special controls, and premarket approval

Table 1: Classification of Medical Device Software [8]

5.2. Software engineering practices for medical devices

5.2.1. Safety-critical software development methodologies

The FDA emphasizes the importance of following recognized standards and guidelines for software development, such as IEC 62304 for medical device software lifecycle processes. Key practices include:

- Implementing a quality management system
- Conducting thorough risk analysis and management
- Establishing traceability between requirements, design, and testing
- Documenting the entire development process

5.2.2. Real-time operating systems for medical devices

For devices requiring real-time performance, the FDA expects manufacturers to consider:

- Deterministic behavior and response times
- Fault tolerance and error handling mechanisms
- Memory management and task scheduling
- Compliance with relevant safety standards

5.2.3. Firmware update mechanisms

The FDA guidance on cybersecurity emphasizes the importance of secure update mechanisms. Considerations include:

- Cryptographic verification of firmware updates
- Fail-safe update procedures
- Documenting the update process and its validation

5.3. Quality assurance and testing for medical device software

5.3.1. Risk management and hazard analysis

The FDA requires a comprehensive risk management process as outlined in ISO 14971. This includes:

- Systematic identification of potential hazards
- Estimation and evaluation of associated risks
- Implementation and verification of risk control measures
- Production and post-production information review

5.3.2. Verification and validation processes

The FDA distinguishes between verification (ensuring the device is built correctly) and validation (ensuring the correct device is built). Key aspects include:

- Software verification at each stage of the development lifecycle
- Validation of user needs and intended uses
- Testing under actual or simulated use conditions
- Documentation of test plans, protocols, and results

5.3.3. Long-term reliability testing

To ensure the ongoing safety and effectiveness of medical devices, the FDA expects manufacturers to:

- Conduct reliability testing appropriate to the device's risk classification
- Implement a post-market surveillance program
- Establish procedures for handling complaints and reporting adverse events
- Conduct periodic review of device performance data

The development and quality assurance of medical device software present unique challenges due to the potential impact on patient safety and the stringent regulatory requirements. The FDA's regulatory framework provides a comprehensive approach to ensuring the safety and effectiveness of medical devices throughout their lifecycle. By adhering to these guidelines and implementing robust software engineering and quality assurance practices, manufacturers can develop reliable, secure, and effective medical devices that improve patient outcomes while meeting regulatory requirements.

6. Case Studies

6.1. Implementation of ePrescribing Systems in NHS Hospitals

A study by Cresswell. [9] investigated the early experiences of implementing ePrescribing systems into National Health Service (NHS) hospitals in England. This case provides valuable insights into the challenges and lessons learned from deploying a critical component of health information technology.

Key findings from the study include:

- Motivations for implementing ePrescribing systems included improving patient safety and organizational efficiency.
- Challenges encountered during implementation included:
  - Insufficient workforce planning and training
  - Inadequate infrastructure and systems integration
  - Difficulties in data migration and management
- Success factors identified were:
  - Effective leadership and support from senior management
  - Engagement of clinicians in the implementation process
  - Adoption of phased rollout approaches

The study highlighted the importance of realistic expectations and thorough planning in health IT implementations. It also emphasized the need for ongoing evaluation and optimization post-implementation to realize the full benefits of the system.

6.2. Implementation of Video Outpatient Consultations

Greenhalgh. [10] conducted a mixed-method study on the real-world implementation of video outpatient consultations across macro (national policy), meso (organization), and micro (individual) levels in the UK's NHS. This case study provides a comprehensive view of the challenges and opportunities in implementing telemedicine solutions.

Key aspects of the implementation included:

- At the macro level:
  - National policy supported the introduction of video consultations
  - However, challenges arose in aligning with existing regulatory and payment models
- At the meso level:
  - Organizational change was required to integrate video consultations into existing workflows
  - Technical infrastructure and support were critical for successful implementation
- At the micro level:
  - Clinicians and patients needed to adapt to new ways of interacting
  - The technology had to be user-friendly and reliable to gain acceptance

The study found that successful implementation required alignment across all three levels. It highlighted the importance of considering not just the technology itself, but also the broader context in which it is implemented, including organizational culture, workflows, and individual preferences.

6.3. Lessons Learned from Both Case Studies

While these case studies focus on different aspects of healthcare software implementation, several common themes emerge:

1. Importance of stakeholder engagement: Both studies emphasize the need to involve end-users (clinicians and patients) in the implementation process.
2. Phased implementation approach: Both cases suggest that gradual rollout allows for learning and adaptation.
3. Infrastructure and technical support: Adequate IT infrastructure and ongoing technical support are crucial for successful implementation.
4. Organizational change management: Implementing new systems requires changes in workflows and organizational culture, not just technology.
5. Continuous evaluation and improvement: Both studies highlight the need for ongoing assessment and refinement of the implemented systems.
6. Alignment with broader healthcare context: Successful implementations consider the wider healthcare ecosystem, including policies, regulations, and existing systems.

These case studies illustrate the complex, multi-faceted nature of healthcare software implementation. They underscore the importance of comprehensive planning, stakeholder engagement, and adaptive implementation strategies in ensuring the successful adoption of new healthcare technologies.

Challenge	ePrescribing Systems	Video Consultations
Technical	Inadequate infrastructure, data migration issues	Network reliability, device compatibility
Organizational	Insufficient workforce planning	Integration with existing workflows
User-related	Lack of training, resistance to change	Adaptation to new ways of interacting
Regulatory	Compliance with prescribing regulations	Alignment with telehealth policies

Table 2: Key Challenges in Healthcare Software Implementation [9, 10]

7. Future Trends and Challenges

The World Health Organization's "Global Strategy on Digital Health 2020-2025" outlines a vision for the future of healthcare technology that has significant implications for software engineering and quality assurance in the healthcare sector [11]. This strategy identifies key trends and challenges that will shape the development and implementation of healthcare software in the coming years.

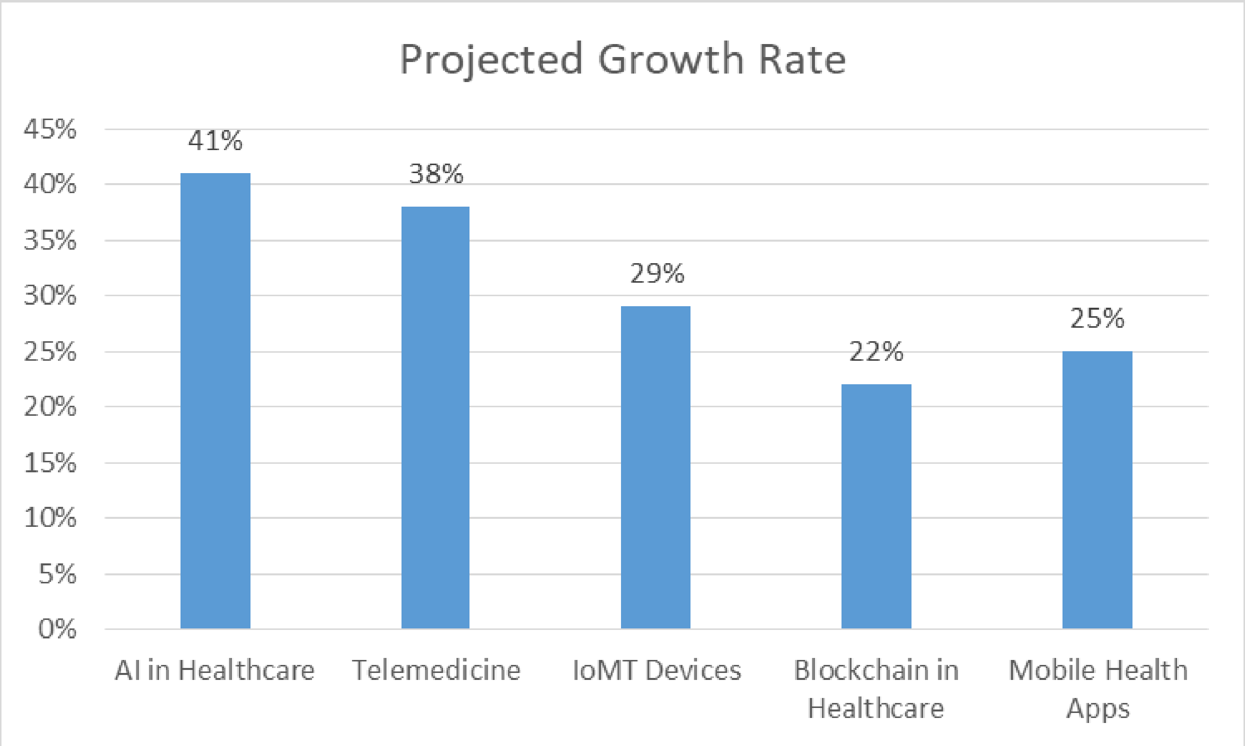

Fig. 2: Projected Growth of Digital Health Technologies (2020-2025) [11]

7.1. Artificial intelligence and machine learning in healthcare software

The WHO report recognizes the potential of AI and machine learning to enhance health systems and medical decision-making. Key areas of focus include:

- Improving diagnosis and clinical care through AI-assisted analysis of medical data
- Enhancing public health surveillance and response through predictive modeling
- Supporting health workforce training and decision-making with AI-powered tools

Challenges for software engineering and QA:

- Ensuring the ethical development and use of AI in healthcare
- Validating AI algorithms across diverse populations to prevent bias
- Developing frameworks for the responsible governance of AI in health systems

7.2. Data management and interoperability

While not specifically mentioning blockchain, the WHO strategy emphasizes the critical importance of secure and interoperable health data management systems. This aligns with the potential applications of blockchain and other advanced data management technologies in healthcare:

- Facilitating secure health information exchange across different systems and countries
- Enhancing data privacy and security in line with legal and ethical standards
- Promoting the use of common data standards and interoperability frameworks

Challenges for software engineering and QA:

- Developing systems that can operate across diverse regulatory and cultural contexts
- Ensuring data quality and reliability in large-scale health information systems
- Balancing data accessibility for research and innovation with robust privacy protections

7.3. Digital health technologies and the Internet of Medical Things (IoMT)

The WHO strategy highlights the growing importance of digital health technologies, including mobile health, telehealth, and connected medical devices. This trend encompasses the Internet of Medical Things (IoMT) and its implications:

- Expanding access to health services through telehealth and mobile health applications
- Improving the management of noncommunicable diseases through remote monitoring
- Enhancing health system efficiency through connected devices and data analytics

Challenges for software engineering and QA:

- Ensuring the reliability and accuracy of diverse digital health technologies
- Developing user-friendly interfaces that are accessible to all populations
- Addressing cybersecurity risks in increasingly connected health systems

The WHO strategy emphasizes that these technological advancements must be implemented in a way that promotes equity, inclusivity, and human rights. It calls for a people-centered approach to digital health, highlighting the need for software solutions that are not only technologically advanced but also ethically sound and socially responsible.

As the healthcare industry continues to digitize and adopt these technologies, software engineers and QA professionals will need to adapt their practices to address these global challenges. This includes developing skills in emerging technologies, understanding diverse regulatory environments, and considering the broader ethical and social implications of healthcare software.

The successful implementation of this global digital health strategy will require close collaboration between technology developers, healthcare providers, policymakers, and communities. It presents both exciting opportunities and significant challenges for the field of healthcare software engineering and quality assurance in the years to come.

Conclusion

The field of healthcare software engineering and quality assurance stands at a critical juncture, poised to play a pivotal role in shaping the future of global health systems. Throughout this article, we have explored the intricate challenges and innovative solutions in developing and maintaining Electronic Health Records, telemedicine platforms, and medical device software. The case studies examined have illuminated the real-world complexities of implementing these systems, highlighting the importance of stakeholder engagement, phased implementation approaches, and continuous evaluation. As we look to the future, emerging technologies such as artificial intelligence, advanced data management systems, and the Internet of Medical Things promise to revolutionize healthcare delivery. However, as emphasized by the World Health Organization's Global Strategy on Digital Health, these advancements must be pursued with a steadfast commitment to equity, ethics, and patient-centricity. The success of future healthcare software will hinge not only on technical excellence but also on its ability to navigate complex regulatory landscapes, ensure data security and interoperability, and ultimately improve patient outcomes across diverse global contexts. As software engineers and quality assurance professionals in the healthcare sector, our challenge is to embrace these emerging technologies while remaining vigilant in our commitment to safety, reliability, and accessibility. By doing so, we can help realize the full potential of digital health to enhance healthcare quality, expand access to care, and improve global health outcomes.

References

1. D. F. Sittig and H. Singh, "A new sociotechnical model for studying health information technology in complex adaptive healthcare systems," Quality and Safety in Health Care, vol. 19, no. Suppl 3, pp.

i68-i74, 2010. [Online]. Available: https://qualitysafety.bmj.com/content/19/Suppl_3/i68

2. K. Kawamoto, D. F. Lobach, H. F. Willard, and G. S. Ginsburg, "A national clinical decision support infrastructure to enable the widespread and consistent practice of genomic and personalized medicine," BMC Medical Informatics and Decision Making, vol. 9, no. 1, 2009. [Online]. Available: https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/1472-6947-9-17
3. U.S. Food and Drug Administration, "Digital Health Software Precertification (Pre-Cert) Program," 2019. [Online]. Available: https://www.fda.gov/medical-devices/digital-health-center-excellence/digital-health-software-precertification-pre-cert-program
4. Office of the National Coordinator for Health Information Technology, "What Are Electronic Health Records (EHRs)?," HealthIT.gov, 2019. [Online]. Available: https://www.healthit.gov/faq/what-electronic-health-record-ehr
5. D. Blumenthal and M. Tavenner, "The "Meaningful Use" Regulation for Electronic Health Records," New England Journal of Medicine, vol. 363, no. 6, pp. 501-504, 2010. [Online]. Available: https://www.nejm.org/doi/full/10.1056/nejmp1006114
6. B. Wosik,., "Telehealth transformation: COVID-19 and the rise of virtual care," Journal of the American Medical Informatics Association, vol. 27, no. 6, pp. 957-962, 2020. [Online]. Available: https://academic.oup.com/jamia/article/27/6/957/5822868
7. A. C. Smith,., "Telehealth for global emergencies: Implications for coronavirus disease 2019 (COVID-19)," Journal of Telemedicine and Telecare, vol. 26, no. 5, pp. 309-313, 2020. [Online]. Available: https://journals.sagepub.com/doi/full/10.1177/1357633X20916567
8. U.S. Food and Drug Administration, "Overview of Device Regulation," 2020. [Online]. Available: https://www.fda.gov/medical-devices/device-advice-comprehensive-regulatory-assistance/overview-device-regulation
9. K. Cresswell,., "Investigating and Learning Lessons from Early Experiences of Implementing ePrescribing Systems into NHS Hospitals: A Questionnaire Study," PLOS ONE, vol. 8, no. 1, e53369, 2013. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0053369
10. T. Greenhalgh,., "Real-World Implementation of Video Outpatient Consultations at Macro, Meso, and Micro Levels: Mixed-Method Study," Journal of Medical Internet Research, vol. 20, no. 4, e150, 2018. [Online]. Available: https://www.jmir.org/2018/4/e150/
11. World Health Organization, "Global Strategy on Digital Health 2020-2025," 2021. [Online]. Available: https://www.who.int/docs/default-source/documents/gs4dhdaa2a9f352b0445bafbc79ca799dce4d.pdf