

Linking Quality Improvement and Health Information Technology through the QI-HIT Figure 8

Trevor Jamieson^{1,2,3} Muhammad M. Mamdani⁴ Edward Etchells⁵

¹General Internal Medicine, St. Michael's Hospital, Toronto, Ontario, Canada

²Women's College Hospital Institute for Health Systems Solutions and Virtual Care (WIHV), Toronto, Ontario, Canada

³Department of Medicine, University of Toronto, Toronto, Ontario, Canada

⁴Li Ka Shing Centre for Healthcare Analytics Research and Training (LKS-CHART), St Michael's Hospital, Toronto, Ontario, Canada

⁵Centre for Quality Improvement and Patient Safety, Department of Medicine, Sunnybrook Health Sciences Centre, Toronto, Ontario, Canada

Address for correspondence Trevor Jamieson, MD, MBI, Division of General Internal Medicine, St Michael's Hospital, 30 Bond Street, Toronto, ON M5B 1W8, Canada (e-mail: jamiesont@smh.ca).

Appl Clin Inform 2019;10:528–533.

Abstract

Keywords

- ▶ implementation and deployment
- ▶ testing and evaluation
- ▶ system improvement
- ▶ outcome and process assessment
- ▶ inpatient care

The implementation of health information technology (HIT) is complex. A method for mitigating complexity is incrementalism. Incrementalism forms the foundation of both incremental software development models, like agile, and the Plan-Do-Study-Act cycles (PDSAs) of quality improvement (QI), yet we often fail to be incremental at the union of the disciplines. We propose a new model for HIT implementation that explicitly links incremental software development cycles with PDSAs, the QI-HIT Figure 8 (QIHIT-F8). We then detail a subsequent local HIT implementation where we demonstrated its use. The QIHIT-F8 requires a reprioritization of project management activities around tests of change, strong QI principles to detect these changes, and the presence of both baseline and prospective data about the chosen indicators. These conditions are most likely to be present when applied to indicators of high strategic importance to an organization.

Background and Significance

Health information technology (HIT) is essential to improving health and health care.¹ The implementation of HIT is a complex interplay of technology, culture, and context.^{2–5} Cultural and contextual variables, such as interruptions, multitasking, task switching, dynamic role redefinitions, and competing interventions can significantly influence the successful implementation of HIT by increasing systemic complexity.^{4,6} These influences are difficult to predict, and may only become apparent after HIT is implemented.^{2,3}

Incrementalism is one way to mitigate systemic complexity. A lack of incremental adaptation is a major reason complex interventions, such as HIT, fail to achieve their

aims.^{6,7} Incrementalism involves progressive modifications to the existing norm, allowing adaption of interventions based on culture and context.^{8,9}

Incrementalism forms the foundation of quality improvement (QI) activity through the Plan-Do-Study-Act cycle (PDSA). QI is a systematic and continuous process that uses serial measurement and intervention to improve the way that health care is delivered. PDSA cycles consist of sequential plans for intervention (Plan), the implementation of those interventions (Do), the assessment of their impact (Study), conclusions about next steps (Act), and then new Plans.^{10,11} PDSAs are not a list of preplanned interventions that one runs in sequence, but rather are single, or bundled, interventions that either improve the system or do not. During each cycle, one reassesses the system,

received

March 11, 2019

accepted after revision

June 10, 2019

© 2019 Georg Thieme Verlag KG
Stuttgart · New York

DOI <https://doi.org/>

10.1055/s-0039-1693456.

ISSN 1869-0327.

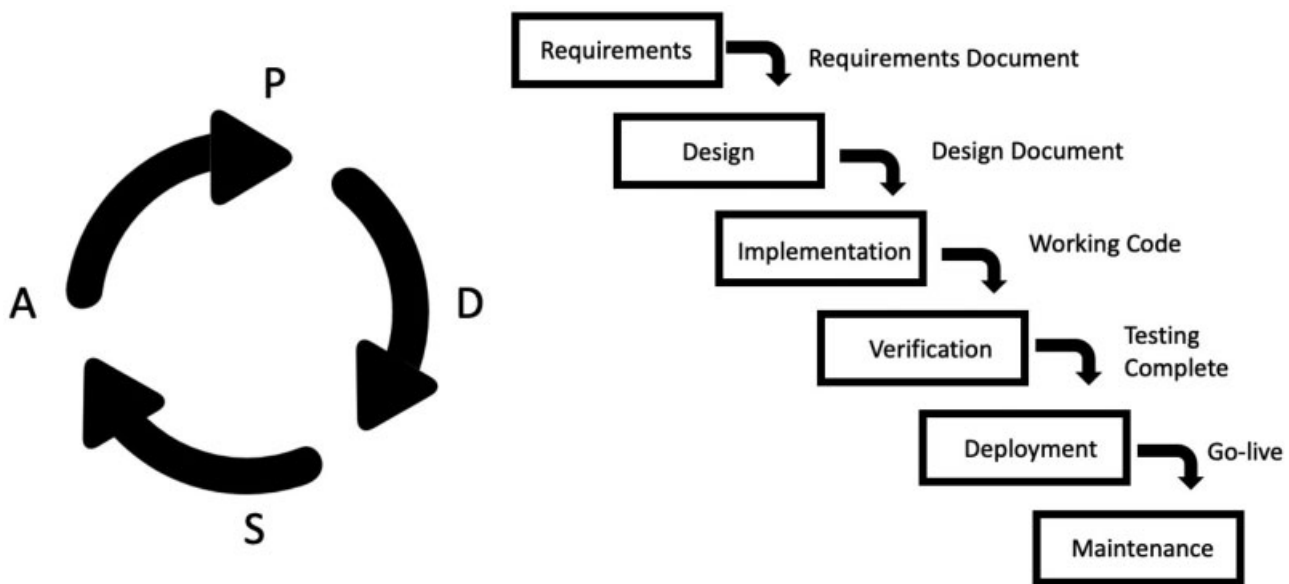


Fig. 1 Plan-Do-Study-Act (PDSA) cycle versus the waterfall software development life cycle. Quality improvement is based on the PDSA, which is inherently cyclical (left). The waterfall model (right) is linear and moves from one stage to the next based on gated outputs ultimately reaching the maintenance stage.

and its shifting context, and determines *new* interventions based on the present state.

Problematically, traditional models of HIT development do not naturally fit into the cyclical model of PDSA (→ Fig. 1). In this case report, we describe a model for more explicit linking of QI and HIT processes, the QI-HIT Figure Eight (QHIT-F8), and demonstrate its use in a project attempting to increase the rates of postdischarge follow-up visits with their primary care providers (PCPs).

The Challenge with Hit as a PDSA Intervention

Traditionally, HIT has been built and deployed using some variant of the “waterfall” software engineering model (→ Fig. 1). This model is a linear, risk-averse model that emphasizes planning. Software proceeds, in stages, through requirements gathering, design, implementation, verification, deployment, and maintenance. You do not move forward until the prior stage has been completed and approved.¹²

Iterative models of development exist. The “spiral” model cyclically proceeds through objectives and risk definition, development and testing, and the planning of the next iteration.¹³ A newer entrant, the “agile” philosophy, places heavy emphasis on customer interaction and feedback.¹⁴ Iterative software development has much in common with related ideas from systems design and innovation like Lean, Lean Startup, and Design Thinking.⁹

In contrast to spiral: (1) agile cycles are much shorter (2–4 weeks vs. 6 months to 2 years); (2) agile assumes that contexts and plans can dramatically change, whereas spiral tracks an overall master plan; and (3) perhaps most importantly, agile focuses on the repetitive delivery of real working software to a customer for feedback, whereas spiral is built around the delivery of iterative prototypes. In the context of PDSA,

“feedback” is less about customer feedback, and more about whether or not a chosen quality indicator is improving because of a deployed intervention, giving an overall advantage to agile as a model for HIT-related PDSA interventions.

Agile models are not completely new to health care. The medical devices industry has demonstrated some cost savings by leveraging agile’s ability to fail fast, succeed quickly, and avoid scope creep.^{15–17} In another use case, an emergency department used agile models to incrementally improve the user experience and workflow in a documentation system.¹⁸ Demonstrations of its value in health care, however, remain rare.

QI experts use many non-HIT strategies, including patient/provider education, organizational change, audit and feedback, provider/patient reminders, facilitated relay of clinical information to providers, financial/legislative incentives, and the promotion of self-monitoring and management.¹⁹ While you could imagine parallel “HIT PDSAs” and “non-HIT PDSAs” in the same environment, it would create complexity and could divorce decisions about non-HIT-related system changes, changes that are often critical for getting value out of HIT innovations,²⁰ from the HIT itself.

A PDSA model incorporating HIT should therefore allow for rapid deployment of incremental HIT, should have the ability to track indicators and determine their movement, and should bring together conventional and HIT-based QI interventions by having QI experts and HIT professionals sit on the same team.

Proposing a New Incremental Model

We propose a model that explicitly links established processes of incremental software development with incremental processes of QI (the QHIT-F8). The model is a figure eight where short software development cycles (“HIT Loops”) dovetail

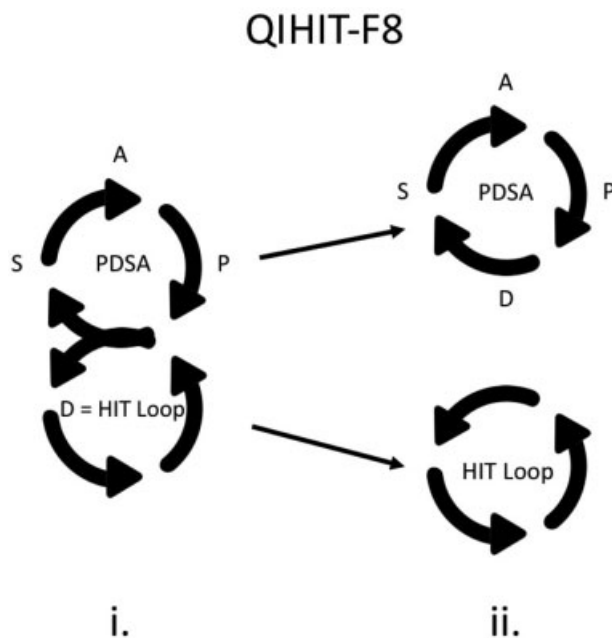


Fig. 2 The quality improvement health information technology figure eight (QIHIT-F8). (i) The continuous loop is a linkage of a traditional Plan-Do-Study-Act (PDSA) cycle (top circle) and an HIT Loop (bottom circle), explicitly linked after a joint prioritization exercise at the “plan” phase of the PDSA. In the first cycle, the loops are coupled as a figure 8 (QIHIT-F8) as the intervention requires software development—the HIT Loop creates the user-focused intervention and the QI Loop assesses the impact. (ii) Not all interventions need be software development in nature; during these cycles, the loops operate independently. The PDSA cycle would focus on a more traditional QI intervention. The HIT team would focus on bug fixes or software enhancements.

directly into PDSA cycles (→ **Fig. 2**) forming an unbroken continuous loop. The primary loop is the PDSA. At the “Plan” stage of the PDSA, a decision is made on whether the next intervention (or “Do”) will be a software implementation or a nonsoftware intervention, such as an educational session or audit and feedback. The decisions at each plan stage are driven by the primary goal of improving a health care process or outcome, and success is judged by said improvement.

If a software intervention, the loops link into the QIHIT-F8 with the “Do” becoming an incremental software development loop that concludes with the provision of new or updated software into the system. If a more traditional QI intervention, the loops decouple. That PDSA is then primarily focused on the implementation of the QI intervention, while the HIT Loop gets an “idle cycle” to perform activities like bug fixes or enhancements targeted to the user experience. These idle cycles may also be used to work on more involved interventions that could take several PDSA cycles to complete.

Our model acknowledges that HIT-driven QI requires users, clinical improvement specialists, and technology creators. However, despite working toward a common QI target, the loops are oriented from slightly different philosophies. The HIT Loops are rooted in the tradition of clinical informatics and embody the principle that software must be usable and functional to be beneficial.²¹ The dominant PDSAs ensure that the

software interventions are ultimately about improving health care quality. All team members (QI and HIT) are involved in all phases of each PDSA cycle irrespective of intervention type.

The Case: Email Notification of Admission

Our rate of patients seeing their PCPs in follow-up after an acute hospital admission, a provincial quality target, was low.²² As many PCPs are not aware of their patients’ admissions,²³ we felt that improving communication between the inpatient care team and the PCP would increase follow-up rates through awareness and involvement. A proposal involved linking the PCP information in our electronic admission documentation system to a secure email address in a provincial directory.²⁴ An admission notification email would be sent to the PCP from the care team, allowing for easy reply and ongoing clinical discussion. The users would be physicians completing admission notes. Due to the provincial directory being federated and nonindexed, automatic matching of identity and email was not possible, and manual search/match was required.

This project was resourced within the preexisting project management infrastructure with processes that we could not drastically reorganize. Our development team had already moved to a more agile approach with daily scrums and sprints, but not the intense near daily interactions with customers that purist agile would require. Project check-ins with stakeholders occurred typically every 2 to 4 weeks. Within this preexisting structure, we inserted a few easy-to-incorporate elements from our model: (1) we tracked a quality indicator prospectively with display and discussion at project check-ins and (2) we included a QI expert on the team to allow for discussion of both technological and nontechnological options for improvement during the next cycle.

Our team included software users/clinicians, developers, an HIT project manager, and a QI lead from our family health team. Our aim was for a notification to be sent in 70% of admissions where the PCP had a secure email in the directory.

We deployed a minimum viable product that allowed the linking of the PCPs’ names in the demographics section and their emails in March 2016. We tracked the percentage of eligible PCPs receiving a notification weekly. We also reviewed monthly unstructured informal feedback from users of the intervention and recipients of the notifications.

We had resources to iterate on the product for 6 months (→ **Fig. 3**).

The QIHIT-F8 team met monthly to review notification frequency data (Study), to assess the need for ongoing intervention (Act), and then to collectively determine the nature of that intervention (Plan). If the next intervention was felt to be a nonsoftware intervention, the loops would decouple and the HIT team would focus on other activities until the next meeting.

The cycles proceeded as follows (→ **Fig. 3**):

- Cycle 1 was a coupled cycle and began at the “Do,” that is, the initial deployment of a minimally viable product. After an initial spike in notifications to 70%, the rate dropped to near 40%. We felt this high initial rate at go-

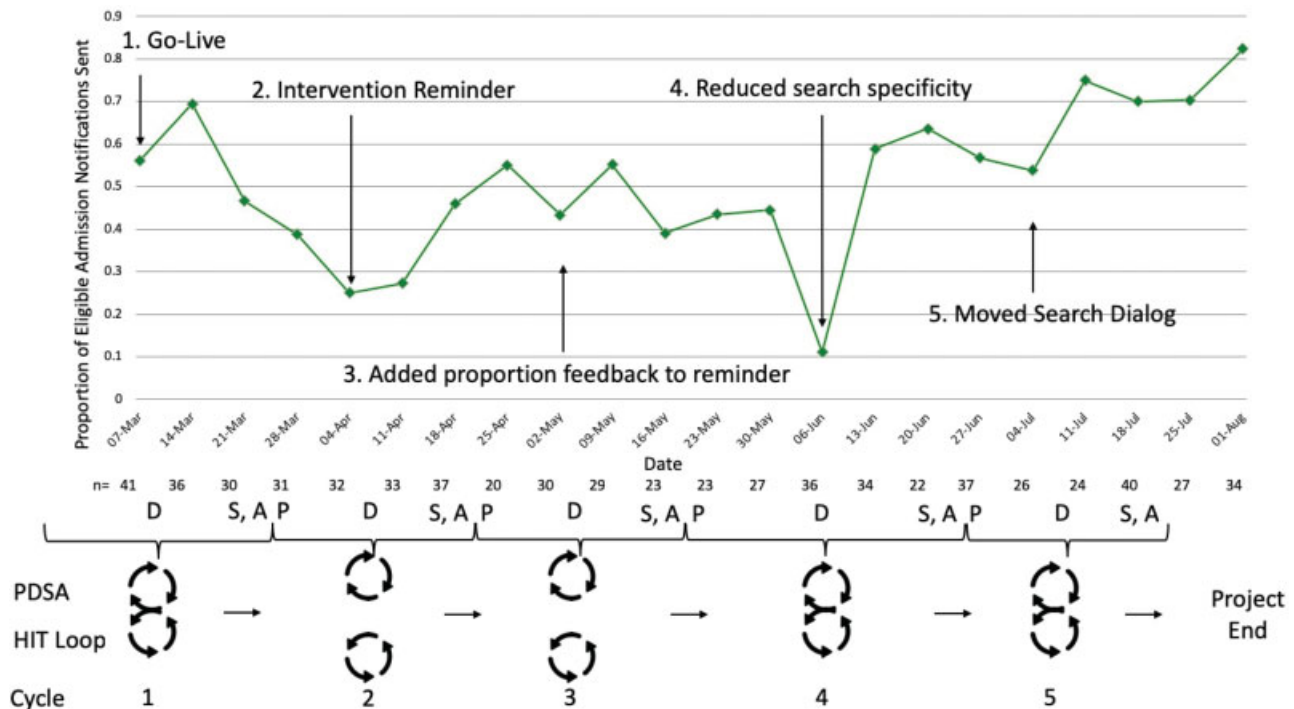


Fig. 3 The proportion of eligible primary care providers (PCPs) sent an admission notification by week of iterative modification of the intervention. Below the graph are the 5 Plan-Do-Study-Act (PDSA) cycles: in cycles 1, 4, and 5, the quality improvement (QI) and health information technology (HIT) Loops are coupled into the QIHIT-F8 as the primary interventions require software development. In cycles 2 and 3, the QI and HIT Loops are decoupled as the primary interventions do not. *n* = the number of eligible primary care practitioners each week.

live was due to excitement driving rapid, yet not sustained, early uptake of the intervention.

- Cycle 2 was a decoupled cycle. The intervention, nonsoftware, was a weekly reminder about the existence of email system. This resulted in some improvement to rates of above 40%. During cycle 2, we learned that our PCP search algorithm was overly restrictive. For example, a search could miss Dr. Jones-Smith if the hyphen was not included. During this idle cycle, the HIT Loop did some minor changes to the wording of the email notification based on recipient feedback.
- Cycle 3 was a decoupled cycle. The intervention, nonsoftware, was audit and feedback: the current notification rates and a reminder of the target of 70% were added to the weekly reminder. The HIT Loop began work on improving the search. Rates at the end of this cycle remained stagnant at 40 to 55%.
- Cycle 4 was a coupled cycle with the primary intervention being the introduction of a less restrictive search algorithm. There was immediate improvement of rates to 60%. During cycle 4, we learned that some users were failing to notice the email search box, which was located in the rarely accessed demographic section of the note.
- Cycle 5 was a coupled cycle with the primary intervention being to duplicate the search box and PCP information in the sign-off area of the note after which rates rose to more than 70% for 4 consecutive weeks.

Lessons Learned

First, this model requires some local control over software development, which is not a given in the vendor-dominant

health care ecosystem. However, we hope that with the greater availability of application programming interfaces like Fast Healthcare Interoperability Resources (FHIR), there will be more flexibility to create homegrown custom applications by clinical sites.²⁵

Second, a culture shift in HIT project management will be necessary. Success, and hence development effort, is defined by improvement in a quality indicator rather than by being “on time/on budget.” While the QIHIT-F8 does not preclude appropriate scoping, we had a limited time to make changes, for example—a challenge is that improvement may not always occur quickly. It is unlikely we could have extended the project significantly had there been very slow improvement over time. Such resourcing is more realistic for a small number of highly strategic indicators where improvement is part of a strategic plan. That inclusion also has the advantage of ancillary system resources to clearly describe the problem and the current state of the system, mitigating the risk that PDSA cycles could be driven by inappropriate “quick and dirty” interventions based on weak understanding of the system.¹¹ Our next step will be to get approval for a more intense application of the model concurrent with an evaluation of the value of the model for an indicator of high strategic priority.

Third, this model requires prospectively available quality indicator data. Originally, we had a secondary aim to improve on the frequency of 7-day postdischarge follow-up rates with the patients’ PCPs. Unfortunately, due to human resource challenges, we could not obtain the 7-day follow-up data prospectively, and so changes in that measure never drove

decision making. However, since 2018, we have access to a hospital data warehouse that is improving our local access to real-time data.²⁶ This is also an advantage of strategic indicators as they are more likely to have been longitudinally audited and tracked.

Fourth, a major challenge may be detecting whether or not a change has occurred. There are well-established rules for detecting change with QI,²⁷ but even these can be challenging to apply—for example, our fidelity measure requires the intervention to exist, meaning there is no baseline upon which to construct a useful control chart. Additionally, there may be a delay between deployment, adoption, and optimal use that has to be considered. These issues are not HIT specific; they are issues QI experts deal with daily, further justifying their inclusion on the team. In our project, we cannot confidently say we increased the rate of email notification as, lacking a strong baseline, our PDSA timeframe was too short to detect real movement. Again, an indicator with an established baseline that allows for a proper control chart with control limits is preferred.

Lastly, the QIHIT-F8 should not necessarily be applied to all software development problems. This model applies specifically to HIT-facilitated QI or to systems that are inherently more complex, and hence less predictable. For highly predictable and specifiable systems, traditional software engineering models based on rigorous requirements gathering and design prior to the creation of any software are still valuable. The nonadoption, abandonment, scale-up, spread, and sustainability framework is one possible way to define the complexity of a health care software intervention or health care domain.⁶

Conclusion

Health care has already embraced incremental philosophies of change, most notably in the field of QI. Increasingly, incremental software development philosophies, like agile, are also being used to deploy software. These development philosophies have the advantage of being able to closely track iterative processes, like the PDSAs in QI. We propose a model where quality indicators explicitly drive the incremental provision of software. This model requires project management reprioritization around the movement of quality indicators as well as baseline and prospective data about those indicators to be available. Both are more likely if the chosen indicators are of high strategic importance to the organization.

Clinical Relevance Statement

Quality improvement is an incremental activity with serial improvements enacted through Plan-Do-Study-Act (PDSA) cycles. Many software engineering and project management activities, being nonincremental, are difficult to apply to quality improvement activities, and can be difficult to explicitly link to the improvement of a health care indicator. We propose a model, the QIHIT-F8, that allows one to link incremental software development to quality improvement

cycles ensuring that health care technology drives toward clinical value.

Multiple Choice Questions

1. An example of an incremental software development philosophy would be:
 - a. PDSA.
 - b. Agile.
 - c. Waterfall.
 - d. LEAN.

Correct Answer: The correct answer is option b. Agile is a software development philosophy proposed in 2001's Agile Manifesto that is based upon the incremental growth of software through frequent customer feedback. PDSA and LEAN are both iterative and incremental philosophies in the domains of quality improvement and process improvement specifically. Waterfall is a phase-gated software development philosophy with multiple stages (Requirements, Design, Implementation, Verification, Maintenance) in which one does not move to the next step until the prior step is completed. It is not incremental in nature.

2. In quality improvement, at what step of the PDSA does one decide on the nature of the subsequent intervention:
 - a. Plan
 - b. Do
 - c. Study
 - d. Act

Correct Answer: The correct answer is option a. Although different sources may report slightly different activities at each step, the original specification of the PDSA would suggest that it is at the Plan step that one analyzes the current state of the system, determines the next best intervention to attempt, and designs that intervention. At the Do step, the intervention is implemented. At the Study step, the current status of the quality indicator one is attempting to shift is assessed. At the Act step, a decision is made whether to continue intervening or whether the indicator has improved sufficiently to stop.

Protection of Human and Animal Subjects

This implementation received a quality improvement waiver and did not require research ethics board approval. This was not considered human subjects research.

Conflict of Interest

None declared.

Acknowledgment

The original implementation of our admission documentation tool and the follow-up enhancements to the system were made possible by the leadership of Anne Trafford, Michael Freeman, and the project management and software development teams at St. Michael's Hospital. The operationalization of the QIHIT-F8 was facilitated by the

project management leadership of Giuseppe Cammisa, and the quality improvement support of Samantha Davie.

References

- 1 Blumenthal D. Stimulating the adoption of health information technology. *N Engl J Med* 2009;360(15):1477–1479
- 2 Sittig DF, Singh H. A new sociotechnical model for studying health information technology in complex adaptive healthcare systems. *Qual Saf Health Care* 2010;19(Suppl 3):i68–i74
- 3 Leonardi PM, Barley SR. What's under construction here? Social action, materiality, and power in constructivist studies of technology and organizing. *Acad Manage Ann* 2010;4(01):1–51
- 4 Cresswell KM, Sheikh A. Undertaking sociotechnical evaluations of health information technologies. *Inform Prim Care* 2014;21(02):78–83
- 5 Desveaux L, Shaw J, Wallace R, Bhattacharyya O, Bhatia RS, Jamieson T. Examining tensions that affect the evaluation of technology in health care: considerations for system decision makers from the perspective of industry and evaluators. *JMIR Med Inform* 2017;5(04):e50
- 6 Greenhalgh T, Wherton J, Papoutsis C, et al. Beyond adoption: a new framework for theorizing and evaluating nonadoption, abandonment, and challenges to the scale-up, spread, and sustainability of health and care technologies. *J Med Internet Res* 2017;19(11):e367
- 7 Cresswell KM, Bates DW, Sheikh A. Ten key considerations for the successful optimization of large-scale health information technology. *J Am Med Inform Assoc* 2017;24(01):182–187
- 8 Slutsky JR. Moving closer to a rapid-learning health care system. *Health Aff (Millwood)* 2007;26(02):w122–w124
- 9 Bhattacharyya O, Blumenthal D, Stoddard R, Mansell L, Mossman K, Schneider EC. Redesigning care: adapting new improvement methods to achieve person-centred care. *BMJ Qual Saf* 2019;28(03):242–248
- 10 Leis JA, Shojania KG. A primer on PDSA: executing plan-do-study-act cycles in practice, not just in name. *BMJ Qual Saf* 2017;26(07):572–577
- 11 Reed JE, Card AJ. The problem with Plan-Do-Study-Act cycles. *BMJ Qual Saf* 2016;25(03):147–152
- 12 Balaji S, Murugaiyan MS. Waterfall vs. v-model vs. agile: a comparative study on SDLC. *Intl J Inform Technol Business Manage* 2012;2(01):26–30
- 13 Boehm BW. A spiral model of software development and enhancement. *Computer* 1988;1(05):61072
- 14 Manifesto for Agile Software Development. Published 2001. Available at: www.agilemanifesto.org. Accessed March 6, 2019
- 15 Montoya M. Agile: a prescription for improved healthcare technology and delivery. *cPrime*. Published July 16, 2012. Available at: <https://www.cprime.com/2012/07/agile-development-in-healthcare-technology-industry/>. Accessed March 6, 2019
- 16 Rasmussen R, Hughes T, Jenks JR, et al. Adopting agile in an FDA regulated environment. 2009 Agile Conference; 2009: 151–155
- 17 Reifer DJ. How good are agile methods? *IEEE Softw* 2002;19(04): 16–18
- 18 Bishop RO, Patrick J, Besiso A. Efficiency achievements from a user-developed real-time modifiable clinical information system. *Ann Emerg Med* 2015;65(02):133–142
- 19 Shojania KG, McDonald KM, Wachter RM, Owens DK. Closing the quality gap: a critical analysis of quality improvement strategies, volume 1-series overview and methodology. Technical review 9 (contract no. 290–02–0017 to the Stanford University-UCSF Evidence-based Practices Center). AHRQ Publication No. 04–0051–1. Rockville, MD; Agency for Healthcare Research and Quality; August 2004. Available at: <https://ncbi.nlm.nih.gov/books/NBK43908>. Accessed April 8, 2019
- 20 Rudin RS, Bates DW, MacRae C. Accelerating innovation in health IT. *N Engl J Med* 2016;375(09):815–817
- 21 Wears RL. Health information technology and victory. *Ann Emerg Med* 2015;65(02):143–145
- 22 Health Quality Ontario. Primary care post-discharge follow-up. Health Quality Ontario's Quality Compass. Published 2015. Available at: <https://qualitycompass.hqontario.ca/portal/primary-care/Post-Discharge-Follow-Up>. Accessed March 6, 2019
- 23 Bell CM, Schnipper JL, Auerbach AD, et al. Association of communication between hospital-based physicians and primary care providers with patient outcomes. *J Gen Intern Med* 2009;24(03): 381–386
- 24 Jamieson T, Ailon J, Chien V, Mourad O. An electronic documentation system improves the quality of admission notes: a randomized trial. *J Am Med Inform Assoc* 2017;24(01):123–129
- 25 Overview FHIR. hl7.org. Published April 19, 2017. Available at: <https://www.hl7.org/fhir/overview.html>. Accessed July 4, 2018
- 26 LKS-CHART. Available at: <https://www.chartdatascience.ca>. Accessed March 6, 2019
- 27 Benneyan JC, Lloyd RC, Plsek PE. Statistical process control as a tool for research and healthcare improvement. *Qual Saf Health Care* 2003;12(06):458–464