A Methodology for Development of Clinical Performance Monitoring Applications

Pilar Mata

Directed By:
Prof. Liam Peyton

Thesis

Submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of

Master of Science in Electronic Business Technologies

uOttawa

University of Ottawa

Ottawa, Ontario, Canada

August 2015

Abstract

Clinical performance monitoring applications enable performance management of care processes in clinical settings. Although information technology has been advocated as a solution to support the provision of better care, the development of clinical performance monitoring applications is often a non-trivial task. A high rate of failure in IT healthcare project implementations has been reported in the literature due to the disconnect between clinicians and the development team. Furthermore, challenges inherent to the configuration of the healthcare system add to the complexity of developments. Often data sources are not adequately structured or cannot be accessed in a timely fashion; processes are uncoordinated or ill-defined; a plethora of information technologies across different healthcare organizations make interoperability problematic; and there are concerns related to privacy and security. Getting the right information to measure the achievement of the right goals at the right time for the right people is the main task to address when developing clinical performance monitoring applications.

In this thesis we propose a development methodology that combines technical and managerial aspects of application development following a user-centered approach. It involves the engagement of stakeholders and users throughout in a three phase iterative process of modeling, implementation and evaluation to ensure user acceptance and adoption of applications when deployed. In particular, our focus is on the development of mobile clinical performance monitoring applications, where raw data about clinical problems are logged by healthcare providers and then transformed into meaningful reports that will support decision-making. The development methodology is evaluated

using a case study of a Resident Practice Profile (RPP) application that was developed by a team lead by Dr. Gary Viner from the University of Ottawa medical school.

Acknowledgements

First, I would like to thank my thesis supervisor, Dr. Peyton, for his patience, guidance, continuous encouragement and support during this research. I have enjoyed and learned so much from our thesis meetings. Your passion for mobile app development to enable performance management for healthcare is truly contagious and it has made this journey a truly rich and rewarding learning experience.

Next, I would like to thank my beloved family: my husband, Daniel, and my kids, Isabella and Tomas, for your love, patience and for being a great support during my studies. You are awesome! To my parents for always being there when I need you and for the best education I could have ever had.

Also, I would like to thank Dr. Archibald and Dr. Kuziemsky for their support and valuable feedback during this research. Our interactions brought different perspectives to the research and have made this work a great interdisciplinary experience. Thank you! Also I would like to thank Dr. Gary Viner, MD, Dr. Susan Humphrey, MD and Dr. Eric Wooltorton and all the people in the laboratory, for the great learning experiences I have shared with each of you: Natalia Castillo, Kavya Mallur, Ashish Chattanni, Austin Chamney, Katherine Cheng Li, Ishan Metha, Pramuhk Rao, Isabella Ferreira and so many others with whom I have worked during my studies.

Thank you to IBM, MITACS, and the Ontario Graduate Scholarship Program for their financial support during my studies.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
ACGME	Accreditation Council on Graduate Medical Education
AR	Action Research
BA	Business Analytics
BI	Business Intelligence
CPMA	Clinical Performance Monitoring Applications
DSR	Design Science Research
EHR	Electronic Health Record
EMR	Electronic Medical Records
ETL	Extract Transform Load
FM	Family Medicine
HIS	Health Information Systems
IT	Information Technology
JIT	Just in Time Medicine
MHS	Mobile Healthcare Systems
OHIP	Ontario Health Insurance Program
OLAP	Online Analytical Processing
ORM	Object Relational Mapping
RPP	Resident Practice Profile
SAID	Standards and Indicators Dashboard
TA	Think Aloud
TOH	The Ottawa Hospital
UI	User Interface

Chapter 1. Introduction

1.1. Research Problem and Objectives

There is an increasing interest in leveraging mobile technologies to monitor performance of clinical processes, healthcare strategies and performance of physicians in the healthcare domain (Baarah, Mouttham, & Peyton, 2012; G. S. Ferenchick et al., 2013; Iglar, Polsky, & Glazier, 2011; Kho, Henderson, Dressler, & Kripalani, 2006; Mata, Kuziemsky, Singh, Baarah, & Peyton, 2014). This is due in part to the increased capabilities of mobile devices and the rapid adoption rate of these technologies by physicians in their daily lives (Kho et al., 2006). Flexibility, portability, and the ability to retrieve information quickly are some of the advantages reported on the use of mobile devices in clinical settings (Wallace, Clark, & White, 2012). A positive attitude towards the integration of mobile technologies in clinical practice was recently reported in a survey conducted among ACGME physicians (Sclafani, Tirrell, & Franko, 2013).

Although mobile technologies offer great capabilities and benefits to users in the health care domain, the development of clinical performance monitoring applications (CPMA) is often a complex task. A high rate of failure in IT healthcare project implementations has been reported in the literature (Ammenwerth, Iller, & Mahler, 2006) due to the disconnect between clinicians and the development team. A research conducted by Kaplan & Harris-Salamone (2009) shows an emerging agreement on managerial issues that impact the success of health care IT projects that include: communication problems related to not having the right people for requirements

gathering, difficulties in understanding and translating the requirements for different project participants (e.g. clinicians, developers, implementers, and management), difficulties in understanding the workflow of projects and poor agreement on what needs to be done, among others. Kaplan and Harris-Salamone also emphasize the importance of understanding the context of use for the applications, as it is difficult to translate general principles in IT innovations across different organizations and particular settings. Additional challenges for the development of CPMA include: data sources that are not adequately structured or cannot be accessed in a timely fashion; under documented data; uncoordinated or ill-defined processes; a plethora of information technologies across different healthcare organizations that make interoperability problematic; and concerns related to privacy and security of applications.

Getting the right information to measure the achievement of the right goals at the right time for the right people is a key task to address when developing CPMA. Development efforts must be also flexible enough to adapt to the constantly evolving needs of clinicians and complexities of the health care system. But perhaps it is more important to ensure applications are developed in such a way that their use does not interfere with the role of physicians. A balance between simple, easy-to-use interfaces that minimize the data collection burden and provide powerful reporting capabilities is needed (Rowley, Gough, Doylend, Thirkill, & Leicester, 2013).

In this thesis we present an application development methodology that combines technical and managerial aspects of application development following a user-centered approach. It involves the engagement of stakeholders and users throughout in a three

phase iterative process of modeling, implementation and evaluation to ensure use acceptance and adoption of the CPMA when deployed. We confirm the appropriateness of our the development methodology using a case study of a mobile resident practice profile application where raw data about clinical problems seen during clinical training are logged by residents and then transformed into meaningful reports that will facilitate decision-making at different program levels (residents, preceptors and program directors).

The main objectives for the research are twofold. First, improve the development process of CPMA through the use of a user-centered development approach that minimize the disconnect between clinicians and development team and maximize use acceptance and adoption of IT innovations. Second, demonstrate the ability of CPMA to log clinical data in such a way that it enables effective real time monitoring of clinical performance.

1.2. Research Motivation and Contributions

The main motivations behind this study are to; (1) leverage user-centered design methods, i.e. user experience walkthroughs, especially think aloud sessions, to systematically evaluate innovations at each iteration of development and ensure user adoption (2) leverage model-driven development to optimize the configuration of the application (i.e. databases, forms and reports), and (3) leverage multidimensional data models and OLAP technologies to support the generation of real-time reports that measure achievement of goals.

The main contributions of this thesis are:

1. An application development methodology for developing CPMA which includes:
    a. Guidelines to define a requirements model for clinical performance monitoring applications in terms of goals, metrics, data sources and adoption criteria.
    b. Guidelines for mapping clinical dimensions to a database optimized for reporting in a CPMA.
    c. The integration of user-centered evaluation methods (i.e. user-experience walkthroughs/think-aloud sessions) and expert checklists to ensure adoption and effectiveness of innovations.
2. A gap analysis and set of evaluation criteria by which development methodologies for clinical performance monitoring applications can be evaluated.
3. A case study of the Resident Practice Profile (RPP) application, to illustrate and confirm how the application development methodology provides effective guidance on how to model, implement and evaluate a clinical performance monitoring app.

The following papers have been published related to this thesis:

1. 1. P. Mata, A. Chamney, G. Viner, D. Archibald, L. Peyton, "A Development Framework for Mobile Healthcare Monitoring Apps", Personal and Ubiquitous Computing, Elsevier, Vol. 19, Number 3, pp 623-633, 2015.

2. P. Mata, A. Baarah, C. Kuziemsky, L. Peyton, An Application Meta-model for Community Care, Procedia Computer Science, Volume 37, 2014, Pages 465-472, http://dx.doi.org/10.1016/j.procs.2014.08.070
3. A. Chamney, P. Mata, G. Viner, D. Archibald, L. Peyton, Development of a Resident Practice Profile in a Business Intelligence Application Framework, 4th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2014), Halifax, Canada, Sept. 2014. ***Best Paper Award *** http://www.sciencedirect.com/science/article/pii/S1877050914010059
4. P. Mata, C. Kuziemsky, J. Bindra, A. Baarah, L. Peyton, "Engineering A Performance Management System To Support Community Care Delivery", Fourth Symposium on Foundations of Health Information Engineering and Systems, FHIES 2014, Washington, D.C., USA, July 2014.
5. Isabella Ferreira, Pilar Mata, Pramukh Rao. (2014). QuickForms: A BI Application Framework to Enable Smarter Healthcare. N/A. IBM Centers for Advanced Studies: CASCON 2014, 2014-11-03 (Poster)
6. Gary Viner, Eric Wooltorton, Susan Humphrey-Murto, Douglas Archibald, Liam Peyton, Pilar Mata (2014). Action research as an approach to evaluate educational technology innovation: The Resident Practice Profile (RPP). N/A. AIME Medical Education Day 2014, 2014-04-04 (Poster)

1.3. Research Methodology and Thesis Organization

We used a design science research methodology, with 2 iterations of Define, Design, Demonstrate, Evaluate, Communicate steps after initial Identify (Peffers,

Tuunanen, Rothenberger, & Chatterjee, 2007). Our evaluation used mixed methods (Douglas, 2008), with a confirmatory approach (Jaeger & Halliday, 1998) to evaluate in the first iteration and a comparative qualitative approach in the second iteration.

In the first iteration, we evaluated the iterative development of RPP2.0 after the fact, and used thematic analysis (Braun & Clarke, 2006) to analyze think aloud sessions. We reviewed architecture changes and expert criteria and confirmed that RPP2 was successful because of all the aspects of our methodology (not just think alouds and user-centered design).

In the second iteration, we evaluated the final iteration of development that resulted in RPP3.0 and compared ourselves qualitatively, using our evaluation criteria, to related work.

After both iterations, we refined our initial proposed development methodology based on what was learned from the evaluation.

We summarize next, the steps we followed for the development of our methodology:

1. Literature review, working sessions with the technical expert, and gap analysis of existing methodologies, including a two year case study to develop RPP that we joined in progress on September 2013.
2. Gap analysis of existing CPMA, including; Just in Time Medicine (JIT) and LogMD (LogMD).
3. An initial version of our theory was developed based on steps 1 and 2.

4. Confirmatory research was followed on the RPP2 development process (2012 – 2013) to validate our development methodology against the existing one year history of the RPP project. This included analysis of expert check-lists, screen shots of RPP versions, and analysis of think-aloud session recordings and application architecture. Thematic analysis was used to analyze verbatim transcriptions of think-aloud-sessions. This confirmatory approach allowed us to gather evidence on the main innovations during the development process and explain the essential aspects of developing CMPAs.
5. Refinement of our initial development methodology based on; results of the analysis for the first year, and a second gap analysis.
6. We used our development methodology to drive the definition of version 3 of RPP (design science research)
7. Interviews with end users of RPP to complement findings from Think Aloud sessions and gather additional feedback about current and future use of the RPP. Interviews recordings were transcribed verbatim and we used thematic analysis to analyze interview transcripts.
8. Refinement of the development methodology one final time based on experiences.

The rest of this thesis is organized as follows:

In Chapter 2, we define key concepts that set the context for this study. We review related development approaches and similar CPMAs. In Chapter 3, we present a detailed analysis of the research problem. We evaluate current approaches that can be used for the

development of CPMA, and define the gap to be addressed by this work. We define the criteria by which CPMA development methodologies should be evaluated. Finally, we present our proposed methodology for the development of CPMA. In Chapter 4, we describe the case study we used to confirm and validate our application development methodology. In Chapter 5, we evaluate our development methodology. Finally, Chapter 6 is a summary of the major research contributions from this work, limitations of the study, and areas where more work needs to be done to complement/strengthen the contributions made by this research.

Chapter 2. Background and Literature Review

In this chapter, we first introduce and define key concepts that are relevant for the understanding of our development methodology and case study. Concepts are divided into three main categories: (1) clinical performance monitoring; (2) technologies and models; and (3) development methodologies and user-centered design. Finally, we present a related work section where we describe the development approaches against which we compare our application development methodology in Chapter 5.

2.1. Clinical Performance Monitoring

In this section we define concepts related to performance monitoring in general as well as in a clinical context.

2.1.1 Performance Monitoring

Performance is "about deploying and managing well the components of the causal model(s) that lead to the timely attainment of stated objectives within constrains specific to the firm and the situation" (Lebas, 1995).

Performance management is the continuous process of reviewing goals linked to business strategy. This process requires close collaboration between managers and employees in order to define and review the goals and the actions that will be taken to achieve those goals at the individual, department or unit level. It involves systematic planning, execution, monitoring and evaluation of goals in order to improve business effectiveness (Dresner, 2008).

The focus of our research is on performance monitoring, defined as the continuous process of measuring and reporting metrics related to the operational execution of the strategy (actions designed to achieve the goals and objectives). Measurements quantify how well a goal or objective is being achieved. When a target or benchmark is defined for a metric, the metric provides insights on the degree to which the goal, measured by the metric, is being achieved and how well a process or task (action) is being performed (Kronz, 2006). Reports that provide individuals and the organization with quantitative data on how the strategy is being executed are key to enable effective fact-based performance management (Dresner, 2008). For example, one strategic goal for a healthcare provider could be to improve the existing services. In order to ensure the achievement of this goal, the provider could monitor metrics related to the utilization of services and waiting times to access each of the services provided.

2.1.2 **Clinical Monitoring**

Clinical monitoring may refer to monitoring performance of healthcare providers (individuals) or monitoring of a care process. The clinical performance of individuals is the result of a combination of factors that affect the way an individual completes a clinical task or function (Khan & Ramachandran, 2012). These factors, as described by Khan and Ramachandran, include among others: personal traits, cognitive skills, motor skills, workplace environment, emotional state, and physical state. For example, the degree by which a resident performs the removal of a foreign body may vary depending on whether he/she is being supervised, knowledge about the procedure, the health condition of the patient, and the place and time where the procedure takes place (Khan & Ramachandran, 2012).

Clinical performance could also refer to performance of a care process. A care process involves the coordination of care services provided by different medical staff (e.g. physicians, nurses, administrative personnel) at a hospital unit, across multiple units or even across multiple locations in the community for the treatment of one medical condition (Middleton, Peyton, Kuziemsky, & Eze, 2009). Similar to performance of any individual, performance of a care process is the result of a combination of multiple factors that impact the outcome of the process. For example, Palliative care (WHO) is a care process that involves the coordination of services from a multidisciplinary team of healthcare providers, across various locations in the community, for the management of symptoms and pain of patients with terminal illness (Mata et al., 2014).

Monitoring of medical education experiences is one area that has attracted the interest of researchers over the past three decades (Renshaw et al., 2014). Clinical logbooks have been used to document clinical encounters, procedures or exams performed by a resident during clinical training. Logbooks could be in a paper-based format or electronic format (Renshaw et al., 2014). Regardless of the format, one common use of clinical logbooks is to track the breadth of clinical conditions seen by residents during training in order to ensure residents have had a comprehensive exposure to the experiences residents will encounter later in the independent practice. Monitoring clinical experiences will also allow residents, supervising physicians and program directors to identify gaps in practice and design appropriate strategies in order to close those gaps.

Logbooks have been also used as a guide for students in their learning agendas (Connolly, A., Davis, K., Casey, P., Keder, L., Pradhan, A., Page, R., ... & Dalrymple, J., 2010; Denton, DeMott, Pangaro, & Hemmer, 2006; Renshaw et al., 2014). In self-regulating professions, such as the medical profession, self-assessment is considered a critical skill clinicians must develop. Self-assessments refer to "a process of personal reflection based on an unguided review of practice and experience for the purposes of making judgments regarding one's own current level of knowledge, skills, and understanding as a prequel to self-directed learning activities that will improve overall performance and thereby maintain competence" (Eva & Regehr, 2007). The relevance of self-assessments is linked to the continuous monitoring of practice and, if one's ability to perform a task can be assessed in the moment (Epstein, Siegel, & Silberman, 2008; Eva & Regehr, 2007).

2.2. Technologies and Models

In this section we identify key technologies and models that have influenced our development methodology.

2.2.1 **Business Intelligence**

Business Intelligence is the process by which raw data is transformed into useful information used to gain strategic and operational insights to support strategic decision-making and improve business outcomes (Duan & Da Xu, 2012). Business Intelligence (BI) systems are software and solutions (technologies) designed for gathering, integrating, consolidating and reporting data to support users in making business decisions (Watson, 2009).

In traditional BI systems data is collected and integrated from various data sources and stored in target data warehouses or data marts following a process known as Extract, Transform and Load (ETL) (Tang Jun, Cui Kai, Feng Yu, & Tong Gang, 2009). This process is run to verify, clean, integrate and aggregate data before transferring it to the data warehouse. Online Analytical Processing (OLAP), data mining, and query processing technologies are used in BI systems to visualize and analyze data and convert it into information and knowledge (Kalakota & Robinson, 1999). The main components of a traditional BI system are depicted in Figure 1.

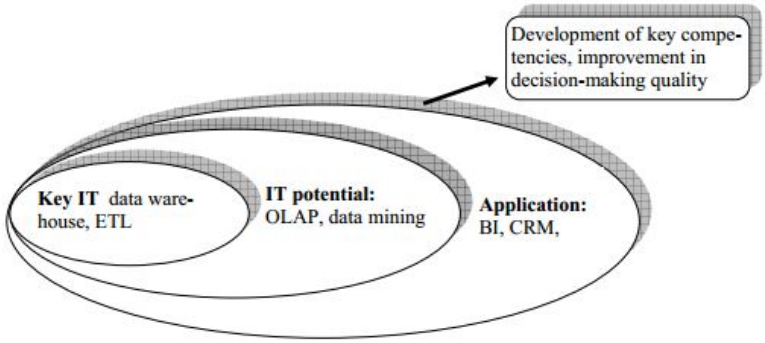

Figure 1: BI Infrastructure Components

Source: (Kalakota & Robinson, 1999)

The use of BI applications is limited when there is a disconnect between strategy and execution, and where the purpose of the organization, department, unit, or individual is not clearly defined and understood (Dresner, 2008). Therefore, the goals and objectives defined during strategic planning must be the basic input for the definition of what needs to be tracked (metrics) by BI systems (Dresner, 2008).

2.2.2 Object Relational Mapping (ORM)

Object hierarchies are used in object oriented applications to hold application data in memory (Tegegne & Peyton, 2013). Object Relational Mapping (ORM) is a technique in which objects are mapped to database tables (Fowler, 2002). An ORM generated database uses a mapping definition to determine the class-to-table or property-to-column mapping in the object relational database. Hibernate (Hibernate) is an example of ORM commonly used in web applications. However, ORM is limited when used for reporting purposes. ORM produces poorly designed databases that require multiple joins between tables when queried thus not optimized for reporting.

2.2.3 Dimensional Data Models

A dimensional data model is the representation of a business in terms of facts and dimension tables (Kimball, 2013). This type of data modeling is reported to be the most effective for BI, as they are optimized for fast querying performance and are easily understood by business users (Kimball, 2013).

Fact tables store numerical performance measurements and dimension tables represent the granularity of facts expressed as textual descriptions of the business. Dimensions are used for querying, grouping, and report labeling and make the data usable and understandable to business users. Dimension tables also represent business hierarchies that are useful for rolling up/down, drilling up/down, and slicing operations in reports (Kimball, 2013). A key advantage of dimensional data models is that new dimensions can be added to the model as long as a single value of the dimension is represented in the fact table. Also, new attributes can be added to pre-existing dimensional tables. This flexibility to accommodate new dimensions with ease is critical

in performance monitoring applications as information needs can rapidly evolve, which requires adjustments to the reporting database (Kimball, 2013).

Dimensional models can be stored in a relational database platform using a star-schema that contains a fact table with measurements linked to multiple dimension tables that resemble a star-like structure (Kimball, 2008).

Online Analytical Processing (OLAP) is a technology that uses multidimensional data structures to store data for fast querying and reporting. OLAP represents the functional and performance requirements of databases targeted for decision support (Chaudhuri & Dayal, 1997). In OLAP technologies, data is organized in hierarchies, or structures that arrange members of a dimension into different levels (e.g. year, quarter, month, week, day), and stored in cubes, or structures that aggregate measures by dimensions, hierarchies and levels used for analysis (Microsoft, 2015). Data organized in such way enable "analysts, managers, and executives to gain insight into an enterprise performance through fast interactive access to a wide variety of views of data organized to reflect the multidimensional aspect of the enterprise data" (Colliat, 1996).

2.2.4 **Technology Acceptance Model**

The Technology Acceptance Model (TAM) (F. D. Davis, 1989) theorized that two particular constructs are key to explain and predict the individual's behavioural intention to use a system. First, perceived usefulness of a system defined as "the degree to which a person believes that using a particular system would enhance his or her job performance". Second, perceived ease of use defined as "the degree to which a person believes that using a particular system would be free of effort." Davis, Bagozzi and Warshaw(1989)

emphasizes the importance of "user acceptance testing" early in the development process of a system as there is more flexibility to change the system without major implications in terms of costs and resources committed at this time of the project. The authors found that, after one-hour hands on introduction, users form perceptions of a system's usefulness and these perceptions are strongly linked to usage intentions that in turn are also highly correlated with the future acceptance of the system.

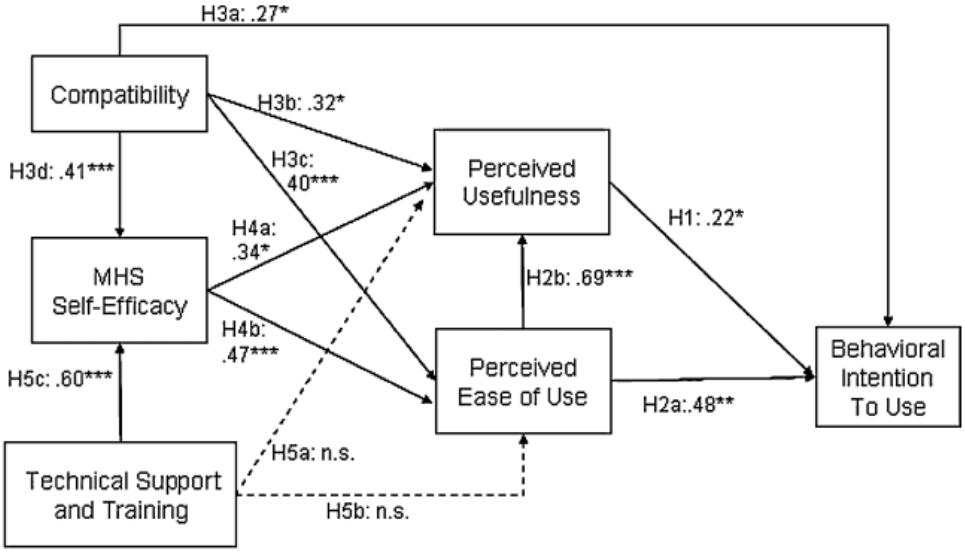

Figure 2: TAM for Healthcare - Empirical Results

Source: (Wu, Wang, & Lin, 2007)

In a domain specific study, (Wu et al., 2007) revised the TAM to explore factors that affect Mobile Healthcare systems (MHS) acceptance. Additional to the two main beliefs that explain behavioral intention of use in TAM (i.e. perceived usefulness and perceived ease of use), Wu et al. found that compatibility is a key construct that explains the intention to use a MHS. Compatibility refers to "the degree to which the innovation is perceived to be consistent with potential users' existing values, prior experiences and

needs". The authors found that compatibility not only directly affects the perceived usefulness, perceived ease of use, and self-efficacy of a system (Figure 2), but its overall contribution to professional's intention to use the MHS is key to successful acceptance of a MHS (Wu et al., 2007).

2.2.5 Electronic Health Records

Gartner defines an Electronic Health Record (EHR) as a system that "contains patient-centric, electronically maintained information about an individual's health status and care, focuses on tasks and events directly related to patient care, and is optimized for use by clinicians. The EHR provides support for all activities and processes involved in the delivery of clinical care" (Gartner, 2013c). EHR are primarily used to integrate paper-based and electronic medical records (EMR) from a variety of sources (e.g. hospitals, physicians clinics, labs) in order to improve quality of care (Gunter & Terry, 2005; Minister of Public Works and Government Services Canada 2010, 2010) EHR facilitates sharing of patient medical records across multiple services providers (e.g. insurance companies, physicians, employers, various departments within a hospital) and provides patients with access to their medical records (Habib, 2010).

2.2.6 Clinical Data Repositories

Clinical Data Repositories are defined as relational databases containing hospital clinical data. According to the Gartner group, a Clinical Data Repository (CDR) is "and aggregation of granular patient-centric health data usually collected from multiple-source IT systems and intended to support multiple uses. When the CDR holds data specifically organized for analytics it meets the definition of a clinical data warehouse" (Gartner, 2013b)

2.2.7 Clinical Performance Monitoring Applications (CPMA)

Clinical Performance Monitoring Applications (CPMA) are a type of BI app used by healthcare organizations to monitor performance of clinical processes, clinical training, and goals of care related to community care delivery. CPMA collects and integrates data from various (re)sources in order to compute metrics to instantiate goals related to clinical performance. Examples of this type of applications include; SAID (Mata, Baarah, Kuziemsky, & Peyton, 2014) and practice profiling application such as LogMD (LogMD,) and JIT (G. S. Ferenchick & Solomon, 2013).

2.2.8 Web Application

A Web Application is a three tier application consisting of a browser-based HTML user interface, a service side application running on the server and a database (Gellersen & Gaedke, 1999).

Java 2 Enterprise Edition (J2EE) is an object-oriented application framework (Johnson, 2005) that can be used to build web applications based on object models.

Hibernate (Hibernate) is an Object-Relational Mapping (ORM) tool that can be used in J2EE web applications to generate a database from the application object model (Singh & Johnson, 2002).

2.3. Development Methodologies and User-Centered Design

2.3.1 Software Development Methodologies

Software development traditionally involves an iterative approach of requirements analysis, design, code and test. More recently, software development methodologies have increasingly incorporated agile methods in order to better address user needs. The

principles emphasized in agile methods are: "*individuals and interactions* over processes and tools; *working software* over comprehensive documentation; *customer collaboration* over contract negotiation; *responding to change* over following a plan" (Beck et al., 2001)

Agile software development methodologies emphasize stakeholder involvement, test-driven development and quick iterations to ensure quality and management of changing or ill-defined requirements (Biju, 2008; Martin, 2003) Extreme Programming (Beck, 2000) and Scrum (Cohn, 2010) are the most common examples of such methodologies.

Behavior-Driven Development (BDD) (Dan North & Associates) is an extension to agile methodologies where users or stakeholders define the expected behavior of the system in detail. Communication and collaboration between development team and users and stakeholders is emphasized so that everyone has the same understanding of what is needed (Sacks, 2012; Soeken, Wille, & Drechsler, 2012).

2.3.2 **User-Centered Design**

User centered design actively involves "users for a clear understanding of user and task requirements, iterative design and evaluation, and a multi-disciplinary approach" (Vredenburg, Mao, Smith, & Carey, 2002). In healthcare domains, where implementation success/failure rates are reported as an important issue (Novak, Brooks, Gadd, Anders, & Lorenzi, 2012) an approach like this is particularly relevant and can be integrated or combined with agile software development methodologies.

2.3.3 Think Aloud Method

The Think-Aloud method is an important technique for incorporating user feedback while evaluating a software system. In this method, participants are encouraged to report verbally their thought process while they are solving a problem (Fonteyn, Kuipers, & Grobe, 1993). The Think Aloud method allows researchers to get insights on the approach and reasoning process users take when solving a task by, (1) identifying what information is relevant to the users and (2) discovering how the information is used during the problem solving process (Fonteyn et al., 1993). As a qualitative method, this method provides rich information requiring a small sample (3+ users) (Holzinger, 2005) for data collection. Verbal reports are recorded and transcribed verbatim for coding and analysis purposes. Thematic analysis (Braun & Clarke, 2006) is conducted to analyze transcripts from Think Aloud sessions.

2.4. Related Works

This section compiles a review of related works that take a different approach to the problem than that of our proposed development methodology.

2.4.1 Electronic Health Records

Electronic Health Records (EHR) is a possible data source choice for clinical performance monitoring. Information about patient encounters (i.e. diagnoses) is recorded in EHRs and these records could be used for reporting. However, some limitations exist. Blumenthal and Tavenner (2010) highlight the fact that, although many healthcare providers acknowledge the benefits of EHRs, its adoption rate is still low. EHRs are mainly available in large institutions and progress towards

implementation in small clinics is slow. According to Skolnik (2011) the criteria used to select the EMR will determine the type of data available in the system. Multiple EMR systems across different settings will result in various data formats. Security policies for data access at different healthcare locations are also a challenge when using EMRs for performance monitoring and learning purposes (Ellaway, Pusic, Galbraith, & Cameron, 2014). Moreover, EMRs does not necessarily contain all the data that maps to the particular performance monitoring information needs of a process or medical task (i.e. medical training). Adjusting EHRs to accommodate the specific performance monitoring needs is often difficult, if not impractical, as reported by the expert in our case study. Other limitations include concerns related to "meaningful use" of EHR systems (Blumenthal & Tavenner, 2010) and skepticism on the accuracy of EMRs for practice profiles (Lyman et al., 2008).

2.4.2 **Agile Methodologies for BI**

Information needs in BI projects are constantly changing. Healthcare organizations have reported to have difficulties in finding cost-effective traditional BI solutions that can be implemented in a reasonable timeframe (Zeng, Jumbo, & Zhang, 2014). In traditional BI projects, data for reporting is transferred from external data sources (e.g. EHR) to the target data warehouse used for reporting. ETL is required to transfer the data to the target data mart, a process that in healthcare environments can be tedious and costly. The complexity of ETL processes in environments such as in healthcare, is often due to: data sources that cannot be easily accessed for privacy and security issues; data owned by multiple services providers (fragmented healthcare data);

non-standards processes for data collection and; data that is not adequately structured for performance monitoring among other issues.

Agile BI has been cited as an approach to enable effective clinical performance monitoring (Zeng et al., 2014). In this approach, BI projects are delivered following an agile software development methodology. Projects are incremental and iterative in nature, following a less formal, more dynamic and customer oriented approach that emphasizes collaboration among stakeholders. This collaborative approach increases accountability, reduces ambiguity of requirements, helps with the understanding of data sources, and ensures quality of results (Kendall & Kendall, 2004; Larson, 2012). In agile BI projects the focus is on the utility of the information delivered and the delivery of working software. Although this approach offers more flexibility to react faster and with greater accuracy to evolving information needs, its applicability in the healthcare domain is limited by issues related to access to the right healthcare data at the right time.

2.4.3 **Custom Web Applications**

Web applications consist of a Web server that generates pages displayed in a browser to collect data that the Web server stored in a database. The database is often generated from the ORM. This approach is used to build custom web applications that records data for clinical performance monitoring. More recently agile methods have been used for the development of custom web applications and the applications are often designed for mobile devices (mobile app).

Just in Time Medicine is a mobile ready-platform developed by clinical researchers from the University of Michigan to facilitate faculty assessments of

learner's clinical skills and for monitoring learner's progress related to clinical problems (G. Ferenchick, Fetters, & Carse, 2008). JIT is Web accessible making the application available across geographically dispersed locations. JIT stores data in a central location, which enables real-time reporting and access to content from anywhere. Students log their clinical experiences, according to the Clerkships Directors in Internal Medicine (CDMI) curriculum, in mobile forms and can see static reports in JIT. Reports can be exported to external programs such as Excel, TIIF, Acrobat reader, etc. The evaluation of the initial version of JIT was done via an online survey after the technology was rolled out to 95 students in 2005. The initial technology was refined and rolled out to 56 students in 2008. The students evaluated the usefulness of JIT (after deployment of the application) by completing an online survey. A significant improvement in the perception of educational utility and technical aspects of JIT was reported, compared to the first version released in 2005 (G. S. Ferenchick et al., 2010).

LogMD is a custom web application developed in conjunction with the Association of Canadian University Departments of Anesthesia. Residents can log their medical experiences in LogMD and compare their performance against national and international benchmarks. Pre-defined reports and graphs in LogMD provide residents with a quick view of their clinical experiences count. The application is cloud based and can be accessed from any Internet enabled device (LogMD).

Chapter 3. Development Methodology

In this chapter we describe our methodology for the development of CPMA that leverages: think aloud protocols and user experience walkthroughs for the evaluation of CPMA; multidimensional data models that support the generation of reports that measure achievement of goals in real-time; and a performance model to ensure the relevance of metrics in reports. The methodology emphasizes a thorough understanding of context of use to ensure user acceptance and effectiveness of the application when deployed.

3.1. Problem Description

Healthcare organizations need to measure performance in terms of quality of care goals (Leggat, Bartam, & Stanton, 2011; Waterson, 2014). Common approaches currently used to monitor clinical performance include; clinical trials sampling, direct observation or ad-hoc reporting at best. Clinical trials sampling is usually expensive and results lack completeness and accuracy (Warner et al., 2013). Direct observation on clinical performance is affected by the ability of evaluators judging the same clinical case with no transparent evaluation standards implemented (G. S. Ferenchick et al., 2013). Ad-hoc reporting, often compiled in loose spreadsheets, is usually done by collecting data towards loosely defined goals or reports that are generated against existing databases that do not necessarily contain the appropriate data to measure performance (i.e. EHR).

The two main tasks involved in clinical performance monitoring are: (1) collect data to compute the metrics that instantiate clinical goals, (2) provide feedback in the form of reports, to clinicians and healthcare organizations, to influence practice and

enable effective clinical interventions. To accomplish the two tasks, we need to ensure that; 1) the data needed to compute metrics can be collected from the available (re)sources and 2) clinicians information needs and adoption criteria are fully understood and incorporated during the development of the application so the technology is useful to clinicians, and not interfere with the day-to-day tasks and procedures of clinical practice; therefore, the technology is accepted, i.e. perceived as easy to use, and useful (Wu et al., 2007).

Figure 3 illustrates a simplified diagram of actors, data sources and data flows for performance monitoring of clinical practice.

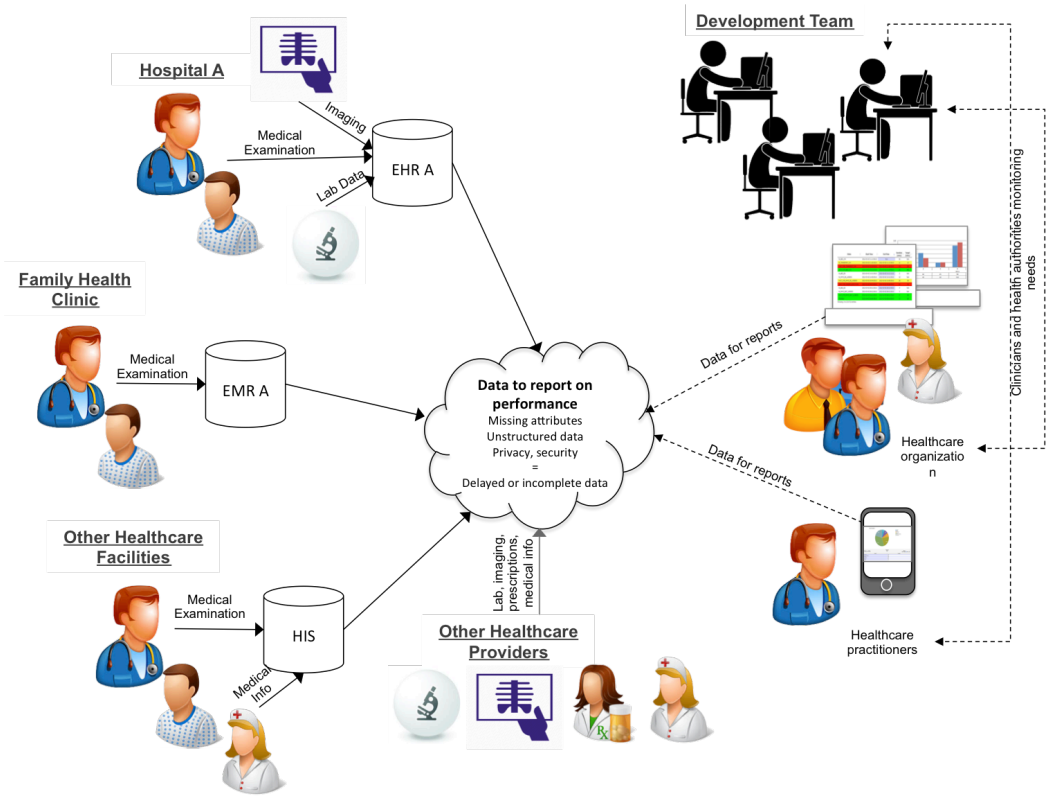

Figure 3: Actors, Data Sources and Data Flows for Clinical Performance Monitoring

Hospitals, family health clinics and any other facility where clinical services are provided (e.g. rural community health centres, home, nursing homes, diagnostic imaging clinics) may have their own Health Care Information systems (HIS) to manage patient clinical records. EHR, EMR or paper based charts are the most common HIS used by healthcare facilities to capture, store and manage patient data. Healthcare providers (e.g. physician, nurses, therapists) could see patients at multiple locations (e.g. residents). Similarly, multiple healthcare providers can treat one patient for one clinical condition at various locations. Each of these interactions generate clinical data that is captured and stored in different HIS, resulting in fragmented, inconsistent or incomplete data that is hard to use for monitoring either, provider's performance or quality of patient care. Therefore, reports on performance will largely depend on the data that can be extracted from different data sources, and quality of data collected (data relevance).

Often, it can be very difficult to extract the necessary data from HIS, and transform it so that it can be used for reporting. In addition, the data may be incomplete, or inconsistent or be structured in such a way that important data for quantifying clinical practice is missing (e.g. free text entry, non-standard definitions, data for billing, claims data). Also, privacy and security issues in healthcare environments often restrict the timely access to data sources.

In addition to technical considerations around data processing, user acceptance and adoption is just as significant a concern. Implementation success/failures rates cited in the literature (Ammenwerth et al., 2006; Heeks, 2005) are often due to a disconnect between clinicians and the development team.

3.2. Gap Analysis

Agile BI and custom web development (which are described in 2.4 Related Works) are approaches that can be used for the development of CPMA. Each of these approaches focuses on one particular aspect of application development (i.e. data sources, user acceptance and adoption, and flexible real-time reports). Also, generating clinical performance monitoring reports from EHR systems could be another option. However, we could not identify in the literature a development methodology that addresses all the aspects of the problem into one single approach that effectively guides the development of CPMA.

Mobile applications such as JIT and LogMd have appeared to overcome challenges related to accessing data that is stored and locked in multiple healthcare databases and/or issues related to incomplete and inconsistent data for performance monitoring. JIT and LogMD integrate clinical data that is needed to evaluate physicians' performance, and generate real-time reports. These applications are typically built following a custom web applications development approach, focused on data collection and not optimized for multidimensional reporting. In custom web development, forms and reports are hard coded and the full functionality of a BI application is limited, i.e. hard to support multidimensional analysis of business data and low flexibility to adapt to changes in information needs. ETL processes are required for the use of third-party reporting tools, which could take days to months to complete. Moreover, custom web developments are IT centric, which are focused on application features rather than usability, overlooking the disconnect between clinicians and the IT development team which is a critical aspect for user acceptance and adoption of IT healthcare deployments.

Agile BI could be seen as one suitable approach to addresses the disconnect between clinicians and the IT development team. As reported by (Gartner, 2013a) BI solutions focus is moving away from "IT-developed reporting solutions toward business-user-led analysis solutions". However, in agile BI the generation of reports is limited to data that is available in and can be accessed from external databases. The availability of the right data to compute the right metrics at the right time continue to be an issue regardless of the user-centric approach in agile BI that closes the gap between business users and the IT development team.

Table 1 summarizes the results from our gap analysis highlighting the limitations of current approaches to develop a CPMA:

Table 1: CPMA - Current Approaches and Limitations

Approaches	**Limitations**
EHR, EMR, other HIS	− Low adoption rates (Fowles & Fund, 2008). − EHR projects in most Canadian provinces are in their strategic plan phase (Minister of Public Works and Government Services Canada 2010, 2010). − EHRs are mainly implemented in institutional settings with approximately 80% of patients encounters occurring outside these settings (Blumenthal & Tavenner, 2010; Fowles & Fund, 2008; Minister of Public Works and Government Services Canada 2010, 2010). − EHR projects are long-term projects. Flexibility to incorporate evolving information needs for performance monitoring is low and costly. − Data incompatibility and data quality issues could result from multiple EHR, EMR (multiple vendors) at different facilities. − Incomplete or inconsistent data for monitoring specific tasks. − Organizational barriers to access EMR and other HIS data sources. − Ill-defined processes that affect data collection in terms of consistency and quality of data. − Skepticism among clinical practitioners on data accuracy for performance monitoring purposes (Lyman et al., 2008; Healthcare Financial Management

	Association (HFMA), 2012) - Data stored in EHR, EMR systems do not necessarily map to performance reporting needs. - Requires ETL processes that can be time consuming and expensive and impede real-time monitoring of processes or tasks.
Agile BI	- Privacy and security issues can restrict access to existing databases. - Not all the data necessary for performance monitoring is available in existing databases (i.e. EHR, EMR). - Requires ETL processes that can be time consuming and expensive and impede real-time monitoring of processes or tasks. - Do not involve end-users in the systematic evaluation of user acceptance and adoption of IT innovations. Testing is done via user cases and user scenarios described by the business user representative. No feedback is obtained from a direct interaction of end-users with the application.
Custom Web Application Development	- Low flexibility to adapt to changing information needs. Hard coded forms and reports. - Focus on the technology aspects of application development: features and functionality. Does not provide a systematic way to evaluate user acceptance and usability. - ETL processes are required to provide full BI capabilities. - Ad-hoc reporting.

3.3. Evaluation Criteria

In this section we define criteria that can be used to evaluate different approaches to building a CPMA. In particular, we will use these criteria in Chapter 5, to evaluate our thesis and compare our approach with the related work we have identified. The evaluation criteria are grouped into three main categories: 1) engineering effort 2) application features and 3) methodology. Each criterion came from one of three sources: literature survey including gap analysis of current approaches; feedback from clinical and technical domain experts who explained what the problem was in current practice; or direct experience of the importance of this criteria during our case study described in

chapter 4. Domain experts consulted during this research include: Dr. Peyton, a technical expert in model-driven data integration and mobile app developments for healthcare (15+ years of experience); Dr. Gary Viner, a clinical expert, director of evaluation from the department of Family Medicine at uOttawa and responsible for monitoring the performance of residents in the Family Medicine program; Dr. Craig Kuziemsky, member of the Standards and Indicators Committee-LHIN, expert in methodological approaches to measure performance in healthcare.

Engineering Effort

The engineering effort measures time, resources and developer skills needed to build, maintain, and enhance one application. The application development methodology and the application framework chosen by the development team, affect the engineering efforts and, in the end, the timeframe and costs in which a project can be delivered. The following criteria set are used to evaluate the engineering effort required when following a specific development methodology.

1. **Form Factors**

This criterion refers to the level of effort required by developers to create, maintain and enhance applications that run in different devices. Application functions should be the same regardless of the device used. Banavar et al. (2000) highlight that a development methodology should focus on the functions supported by the application. The functionality of an application should not be limited to a specific device. Also, the fact that there is a growing trend in the use of mobile technologies among clinicians

(Wu et al., 2007) make of this criterion highly relevant for development of CPMA, as applications should adapt to any platform used.

Engineers and domain experts emphasized the importance of this criterion, as users should be able to choose the device of their preference to run the CPMA rather than being limited to a specific device to use the technology.

2. Form/Report Linkage

The development effort to create forms and reports in a CPMA can be dramatically reduced if forms and reports are linked together by the same database model. Dimension tables populate field values in a form (i.e. drop downs and checkboxes). On the report side, same dimension tables are used to report metrics at different levels of granularity, for grouping, labeling and filtering data. This ensures consistency between data collection and reporting (i.e. data collected in forms is visualized in graphs and tables reports) and reduces the effort needed by a developer to configure forms and reports as both are linked through and defined by the same reporting database. The software technical expert determined the importance of this criterion.

3. Third-Party Reporting Tools

Third-party reporting tools must be supported for in depth analysis to improve family medicine program. This was identified as an important requirement by both Dr. Viner and Prof. Kuziemsky.

4. Application Configuration

This criterion refers to the ease with which a developer or administrator can maintain or update a CPMA. Performance monitoring information needs can quickly evolve. CPMA applications that are hard to configure are more likely to be abandoned by users as applications become obsolete while users wait for the IT development team to deliver new application features that meet their information needs. Ideally, end-users (i.e. administrators) should also enjoy a certain degree of autonomy in the configuration of the application. For, example, add a new value to a drop-down list on a form or create a query to be used for a new report. Both the technical and clinical experts confirmed the importance of this criterion.

5. Repeatable (Defined Method)

This criterion refers to the extent by which two development teams, with the same technical skills, will require the same level of effort to build the same CPMA if they follow one given application development approach. According to (Gartner, 2013a) this criterion is important, as the trend in software development will be to increase the adoption rate of repeatable solutions by different organizations by delivering "packaged domain expertise and applications to enable self-service". Information needs in BI projects evolve quickly and a specific BI tool could easily be out of date quickly. Therefore, it is important that a methodology could be repeatable and packages domain expertise in a structured way in order to accommodate new information needs and minimize the effort required by the development team.

6. User-Centered Design

In the healthcare domain many actors could be involved in a single process. Actors can be external to the organization (i.e. regulatory agencies, other healthcare facilities) or internal to the organization (e.g. doctors, nurses, administrators), each of which analyzes the same set of information differently (Mettler & Vimarlund, 2009). Besides competing information needs between multiple actors, often there is a disconnect between business users (e.g. clinicians) and the development team (IT-centric approach) that could help explain the success/failure rates reported in the literature (Ammenwerth et al., 2006; Forrester, 2015; Heeks, 2005). Therefore, it is important to determine whether the user-centered design is compatible with a specific application development approach in terms of the engineering effort. The importance of this criterion is also emphasized in a report by Forrester(2015) that highlights the success of BI projects keeps a direct correlation with the business ownership in the project.

Application Features

1. Data Entry

This criterion relates to the effort, measured in time, required by the user of a CPMA to complete one record. The clinical domain expert, Dr. Gary Viner, determined the importance of this criterion. Clinicians' routines are very busy and data entry requires additional time commitments by clinicians. As the domain expert expressed, one of the greatest barriers to data collection is the time and energy

required. Therefore, this is an important criterion to use when developing CPMA and evaluating application features.

2. Diagnosis Selection

One critical aspect that affects the intention to use a technology in healthcare is the perceived ease of use of the application (Wu et al., 2007). Diagnosis selection refers to the easiness with which a user of a CPMA can navigate through all possible clinical diagnoses in the application in order to select the right value. The domain expert in our case study determined the importance of this criterion given the possible number of diagnosis relevant to a family patient visit could easily reach ~500 diagnoses.

3. Data Display

Data display refers to the way data is organized and displayed in a form. When too much information is presented in a form, users can easily get lost, particularly when using mobile devices (i.e. small screen sizes) needing to scroll up and down to review values entered for each field. The importance of this criterion was determined through the analysis of think aloud sessions in our case study and the domain experts review.

4. Drill-Through Reporting

Drill-Through reporting refers to the flexibility end-users have to analyze data at different levels of granularity. The importance of this criterion was determined

through the analysis of think aloud sessions in our case study that revealed users considered easy feedback in the form of drillable reports as one of the most relevant factors for user adoption.

5. Near-Real Time Reporting

Near-real time reporting refers to the availability of reports on data collected while a process or task is taking place. The purpose of continuously monitoring a task or process is to support quick and fact-based decision-making. The importance of this criterion was established based on our work with the healthcare expert, Dr. Craig Kuziemsky, for the development of performance monitoring applications for community care. Also, the clinical domain expert determined that, this criterion is important as near-real time reporting could increase the perceived value of the application.

Methodology

1. Clinical Performance (Goals, Metrics)

The utility of a performance monitoring application will largely depend on whether what is being monitored is within the context of a broader system design (goals) and whether changes can be introduced in practice as a result of gaps identified through the monitoring of the execution of a task or process (Dresner, 2008). Before addressing technology needs, first an organization need to understand what goals are going to be monitored and what metrics will be used to instantiate those goals in order to provide users with the information they need to manage their business. This

criterion is used to assess whether a specific application development approach for CPMA integrates a performance model that guide project stakeholders on how to define feasible metrics based on available data sources and criteria for adoption in order to enable continuous performance monitoring.

2. Clinical Acceptance (Adoption Criteria)

A successful application deployment will largely depend on factors that affect user acceptance of the application. Wu, Wang and Lin (2007), in their revised technology acceptance model for Mobile Healthcare Systems (MHS), describe some of these factors, i.e. compatibility, self-efficacy, perceived usefulness and perceived ease-of-use. Although privacy, security, and information quality are not included in Wu, Wang and Lin's model, the authors acknowledge they may also affect acceptance of the technology.

This criterion is important in that, the degree by which a development methodology provides developers with a systematic way of reviewing and incorporating clinical acceptance criteria during the development process will likely affect the behavioral intention to use the technology by clinicians.

3. Model Driven (Reporting Database and Clinical Dimensions)

To measure clinical performance, it is important that the CPMA stores data in a reporting database optimized for performance management and based on an OLAP data model or star schema. In particular, the database needs to map to the goal model and metrics for the CPMA, and should have built-in support for clinical dimensions

(e.g. Diagnoses). This is well established in the literature on performance management (Kimball, 2013).

4. Integrate Forms, Reports and Dimensions

This criterion is closely related to the Forms/Reports Linkage in the Engineering Effort section. The fact that forms and reports are linked through by the same reporting database affects the level of engineering effort required by the development team. Therefore, when selecting a development methodology for CPMA it is important to consider whether there is a systematic way to map clinical dimensions to the reporting database and link forms and reports through the reporting database. Similar to Forms/Reports Linkage, the relevance of this criterion was established by the technical expert.

5. Integrate Clinical Expertise

According to Wu, Wang and Lin (2007) compatibility, or "the degree to which the use of Mobile Health Systems (MHS) is perceived to be consistent with health-care professionals' existing values, prior experiences and needs" is an important factor that impact perceived usefulness and perceived ease of use of a system and in the end, the behavioural intention to use the application. According to the authors, this factor is one of the strongest determinants to user's acceptance of the technology. Therefore, the degree by which clinical expertise (i.e. insights into the previous or current practice processes) is integrated in the application development process is considered an important criterion to ensure application acceptance and adoption.

6. Systematic Evaluation of Usability

Usability or ease of use of an application is been cited by Wu et al. (2007) as one of the factors that affect the behavioural intention to use a technology. It is not only important to ensure an application is technology faultless but also it is important to ensure the technology is accepted and adopted (Benbunan-Fich, 2001). Therefore, this criterion is used to evaluate whether a development methodology include a systematic way to evaluate usability during the application development process.

3.4. Overview of the Development Methodology

In this section we present our three-step methodology for developing clinical performance monitoring applications. The development methodology we are proposing is based on:

- User-Centered design, in which end users and stakeholders are engaged throughout in the development process to facilitate the understanding of adoption and usability criteria and improve the communication between clinicians and the IT development team.
- Flexible and iterative development cycles to adapt to constantly evolving information needs.
- A performance model: metrics linked to goals and data (re)sources and refined by adoption criteria to drive the implementation phase:
    - Report configuration based on metrics linked to goals.

- Form configuration based on a minimum set of attributes required to compute metrics linked to goals.

In our user-centered design approach, users and stakeholders are engaged throughout in an iterative process of application modeling, implementation and evaluation to ensure user acceptance and adoption of the CPMA (Figure **4**). Although all stakeholders participate in each of the development phases, we could say clinical experts lead the modeling phase, engineers lead the implementation phase and end-users drive the evaluation phase.

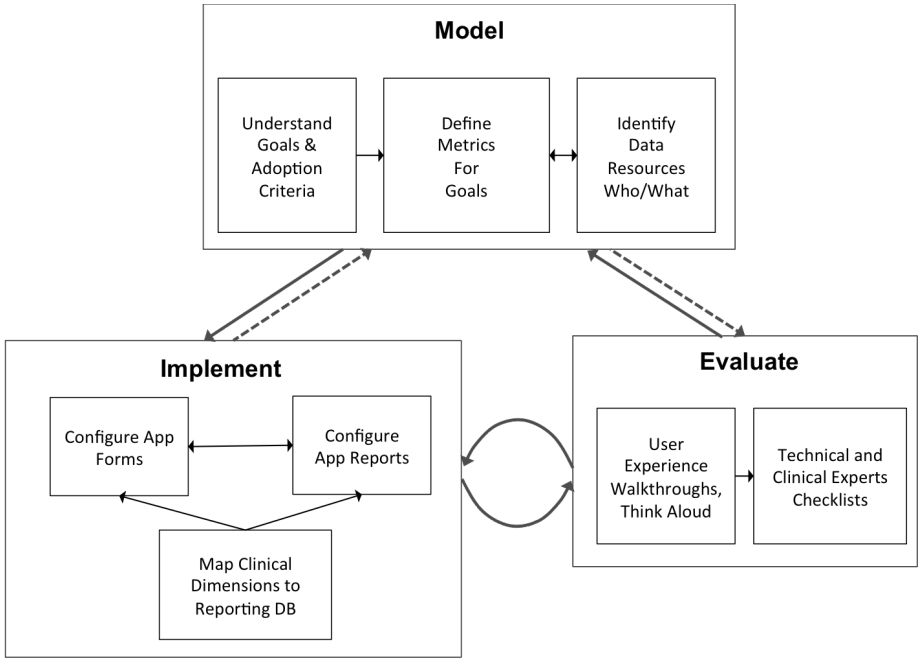

Figure 4: Development Methodology for CPMA

The development of a CPMA starts from the evaluation of clinical practice. Feedback from users is incorporated into engineering and domain experts' adoption checklists is used to model how clinical practice must be monitored. The model is used

to implement a CPMA that collects data in mobile forms to continuously monitor the status of clinical practice through reports. Each implementation phase is followed by a throughout evaluation of the new version of the CPMA while the application is still under development. Stakeholders engage in usability testing and experts conduct trade-off analysis of innovations in order to ensure issues that affect user acceptance and adoption can be addressed. This cycle of Evaluate, Model and Implement is repeated until no significant innovations and barriers to user acceptance and adoption are identified during the evaluation. The end of this cycle is reached when clinical users and the IT development team are in sync and only minor application adjustments are required.

3.5. Model Phase

In order to monitor performance, it is important to have consensus on the goals and metrics used to measure performance amongst the key stakeholders (Mata et al., 2014). Along with the understanding of goals, metrics and data sources, it is also important to understand the context of use that will determine adoption criteria. Figure 5 depicts a generic performance model for CPMA.

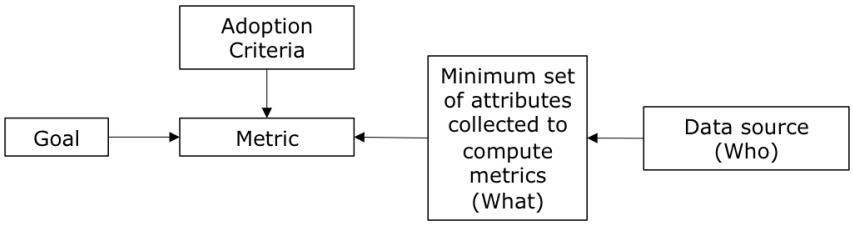

Figure 5: Generic Performance Model for CPMA

3.5.1 Understand Goals And Adoption Criteria

This step consists of understanding what the desired goals for healthcare performance are. At a high level, clinical performance monitoring goals can be grouped into three main categories: education, access and care. Education goals refer to level of certification or training a healthcare provider possesses in relation to the task or process being executed and monitored. Access goals refer to the extent of use of a particular healthcare resource for the provision of care related to the process or task monitored. Care monitoring goals are related to the quality of care provided by the actors/processes that are being monitored. A clinical performance monitoring application could include goals in all of these categories or just one.

In order to ensure user acceptance and adoption of the CPMA, it is also important to identify and define the adoption criteria. Information on; how the application is going to be used, by whom, environmental and user's constrains, frequency of data entry and expected data entry effort (i.e. time to complete a single record or group of records), persistent/non persistent data and, reporting frequency are some examples of adoption criteria reviewed during this phase.

3.5.2 Define Metrics For Goals

In this step of our application development methodology one defines what metrics can be used to quantify and communicate how well the clinical goals are being met. Metrics are better defined if they are linked to goals and, a metric is useful only if, as a result of monitoring it, we can introduce changes in clinical practice. In defining a metric, we need to identify the attributes that will be used to compute the metric. If data

collection for a specific metric is not feasible; we may need to adjust the metric. Along with the definition of metrics, we also need to define benchmarks of targets for each metric (if available) and the level of granularity required (e.g. date, year, month, week).

For example, one goal of a palliative care program may be to identify all palliative care patients and ensure that they are all referred to appropriate palliative care services in order to receive proper treatment (Mata et al., 2014). One metric related to this goal could be the percentage of palliative care patients referred for palliative care treatment in the community (i.e. total palliative patient referrals/total palliative care patients). This metric is useful and relevant only if the palliative care program administrator can take actions to maximize the number of referrals in the community (e.g. design a program to increase the awareness among physicians of palliative care services in the community). Moreover, it is relevant if the attributes needed to compute the metric could be collected from the available resources.

3.5.3 Identify Data (Re)Sources Who/What

Once a metric has been defined for a goal, we need to identify who will collect what data, as well as how and when, in order to compute the metric. In our development methodology, we propose the use of simple, easy-to-use forms to collect the data necessary to compute a metric. Therefore, it is critical to just define the minimum set of attributes needed to compute the metric in order to minimize the data-entry burden on clinicians.

3.6. Implement Phase

Our development methodology leverages the use of QuickForms (Baarah, Kuziemsky, Chamney, Bindra, & Peyton, 2014) an application framework optimized to collect data directly into a reporting database in order to bypass the complexity, effort, and processing delays associated with ETL. The database model we propose for our application development methodology is a multi-dimensional model, i.e. star-schema, with one fact table and multiple dimensions linked to the fact table.

3.6.1 Map Clinical Dimensions to Reporting Database

In this step we configure the reporting database using a generic, standardized star-schema with fact tables and dimensional lookup tables. Each clinical task is mapped to a fact table linked to multiple dimensional tables. Clinical dimensions, relevant to the task or process being monitored, are mapped to dimensional tables that store the textual descriptions of attributes that are used to generate reports (metrics) and populate data fields in a form. If clinical concepts are repeated in two or more dimensional tables, we should consider consolidating them into one single dimensional table with multi-hierarchy columns.

For example, in a palliative care program, program administrators may need to monitor information about patient referrals by healthcare providers, facilities, diagnosis, age and gender. Patient referral is the task being monitored. Healthcare providers, facilities, diagnosis, age and gender are the clinical dimensions linked to the task. In the reporting database, we map patient referrals to a Fact table and providers, facility diagnosis, age and gender are mapped to dimension tables (Figure 6).

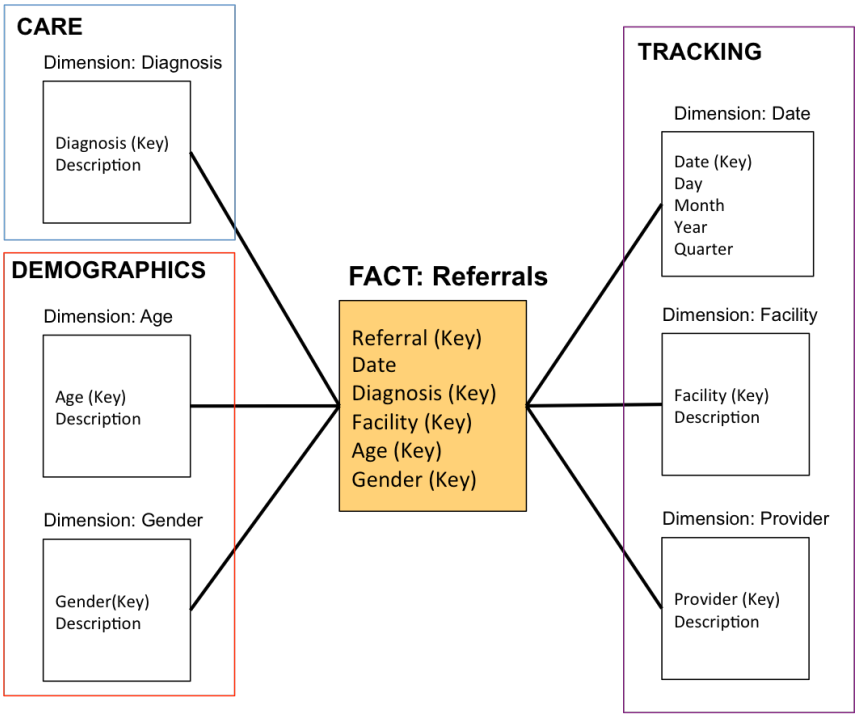

Figure 6: Clinical Dimensions to Reporting Database. Example

At a high level, clinical processes or tasks involve three main clinical dimensions: tracking, demographics, and care. Other dimensions may include education, as this is a critical aspect in the medical profession and affect the quality of care delivered. Tracking dimensions contain data related to where, when and by whom the healthcare service is provided (e.g. date, location, type of care provider). Demographics include characteristics of type of patient such as age, gender, type of patient and any other data related to the patient. Care dimensions contain data related to the care of patients (e.g. assessments, procedures, therapies). Education includes training a healthcare professional has received relevant to the process or task being monitored or any attributes related to the skills of a healthcare provider (e.g. self-assessments, specialty training).

3.6.2 Configure Application Forms

Each dimensional table is mapped to one control in a form. Values stored in the dimensional table supply the values for the controls. This configuration ensures that; the data needed for the computation of metrics is collected in a structured way and, the values in forms controls and associated reports are the same. For example, the dimensional table “Age” is used to populate values in the form control “Age”. The same dimensional table values are used to report age metrics. Attributes with four or more possible values or when multi-selection is available are grouped in multi-select drop down controls. Checkboxes are used for attributes with just a few possible values (e.g. less than four). Text boxes are used to document any additional information; however, these values are typically not used for the computation of metrics.

3.6.1 Configure Application Reports

Metrics in the performance model are mapped to reports. The values from the dimensional table used to populate dropdown values in a form control, are the same values used to compute metrics in the report associated with the form. This ensures consistency between data collected and data reported. Also, values in dimensional tables are used for grouping, labeling and filtering of data in graph and chart reports associated with the metric.

3.7. Evaluate Phase

The evaluation phase consists of user experience walkthroughs using think aloud sessions, as well as technical and clinical expert's checklists that mediate development

efforts in order to ensure that adoption criteria is met at each iteration of development. This phase is key to ensure user acceptance and adoption of the CPMA.

3.7.1 User Experience Walkthroughs (Think Aloud)

At each iteration of development, User Experience walkthroughs using think aloud sessions are conducted to obtain feedback on usability of the CPMA. Typically, participants will include: clinicians championing and managing the project; end-users and; members of the IT development team. User Experience walkthroughs are used to elicit user feedback and gain insights on the thought process of users by observing them trying out the CPMA. Members of the development team observe users and take notes on verbalizations. The participation of clinicians during these sessions is key as they will later generate, in collaboration with the technical experts, the checklists used to guide the development efforts. Sessions are recorded and transcribed for formal analysis of the development process. Notes from the TA sessions are used to (re)define technical and clinical expert checklists used to drive the next iteration of application development. As a result, new requirements and/or adoption criteria could emerge during this phase.

The use of these methods (think aloud sessions and user experience walkthroughs) during the evaluation phase allow the development team and clinicians to: 1) obtain immediate feedback on user's experience and discover relevant areas of improvement; 2) prioritize development efforts towards functions perceived as most useful; and 3) discover key features for adoption.

3.7.2 **Engineering and Domain Experts Checklist**

Experts review the feedback obtained from User Experience walkthroughs and think aloud sessions to prioritize development efforts. The checklists compile results from User Experience analysis, systematic and objective analysis of usage and performance of the CPMA, and technical review, by the technical team, of the application performance (e.g. how many clicks users take to complete a record, functionality check-ups).

Chapter 4. Case Study: RPP (The Resident Practice Profile)

We use a case study of a CPMA for medical training to confirm the methodological steps outlined in our development methodology are helpful to explain key innovations and problems encountered during the development of the application and that, the development methodology is useful to guide the development of similar CPMA.

4.1. Overview

RPP is a mobile CPMA that "enables residents in the Family Medicine (FM) program to self-assess how well their clinical experience, under the supervision of a physician supervisor, provides breadth of experience across the types of patients, diagnoses and procedures that are covered in the postgraduate Family Medicine curriculum." (Mata, Chamney, Viner, Archibald, & Peyton, 2015).

During clinical training, residents see patients across different healthcare settings while supervised by a licensed physician. Although training seeks to provide residents with comprehensive exposure to the main clinical problems and types of patients, in practice, it is difficult for a resident to gain experience in all of the possible diagnoses and procedures relevant to a family medicine patient (~500) during the two years of training. Therefore, a monitoring tool that allows residents, supervising physicians and program directors to track resident's clinical experiences becomes necessary in order to identify gaps in practice and support the design of effective strategies (e.g. self-directed learning) to compensate any deficiencies encountered during this process.

RPP was developed by a group of engineering students at the University of Ottawa, in collaboration with healthcare researchers and doctors from the department of family medicine at the University of Ottawa. Participants include; one technical expert (software engineering professor), three to six engineer students, two doctors (family medicine curriculum director and evaluation director), one scientist researcher and four family medicine resident trainees (two from each year). The project started in September 2012, and went through three complete development iterations, until June 2013, during which significant improvements to the CPMA were obtained.

The initial purpose of RPP included the generation of real-time graphical summary reports of resident's clinical experiences. The original data sources for reports included data from the EHR system at one of the teaching hospitals combined with manual-entry data. The main intent of the reports was to support self-directed learning and planning of clinical practice.

Our research project began in September 2013 when we joined the RPP team. The development of the application was progressing at a good pace and the research team seemed to be successful in what they were accomplishing. We gathered insights on their methodological approach and saw the opportunity to formalize what they were doing in a development methodology. Particularly, because IT healthcare developments have to deal with the inherent complexity of the healthcare system and, in general, available methodological approaches fall short in addressing this issue. Therefore, our aim was to capture, in a development methodology, the main aspects that led RPP researchers to a successful delivery of the application. For this purpose, we got familiar with the

application and gathered insights from the engineering research team (technical expert and students) on the application development process. We conducted a literature review on applications used for purposes similar to RPP and did a comparative analysis of these applications against RPP to determine whether their approach was better than the one followed for RPP. We evaluated if any of these applications (LogMD and JIT) could be used to effectively monitor the training of family medicine residents at uOttawa.

A second analysis was conducted that included a literature review on data sources for medical training performance monitoring applications; particularly the use of EHR systems, as the initial intent for RPP was to combine EHR data with manual data entry for graphical reporting. We wanted to see how it was to build a CPMA from an EHR.

In the next sections we will confirm our methodological steps, at each iteration of development for RPP.

4.2. Initial Version

The initial version of RPP, RPP1, was developed as a custom web app with ORM database and was problematic and not well-received for many reasons including disconnect with clinicians, difficult to update, and lack of integration with clinical dimensions.

4.2.1 **Model**

Goals and metrics for RPP1 were vague and the adoption criteria too broad. Researchers wanted to track resident's exposure to clinical problems by demographics

and diagnoses. As for the adoption criteria, clinicians wanted the application to be mobile accessible, and easy to use. Also, they wanted the application to provide residents with centralized graphical reporting; however, there was not a clear thought of what reports were needed which led to an unsuccessful configuration of RPP forms.

Data sources proposed for RPP1 included a combination of manual data entry and data from an EHR. After a careful review of the EHR system by the technical and clinical experts, they realized that, extracting data from the EHR was not practical for several reasons; diagnoses codes in the EHR were mainly used for billing purposes and did not necessarily map to the family medicine curriculum. Moreover, there was no common EHR to all the clinics where residents practised. Therefore, the data collection mechanism evolved from using EHR records to logging all data in mobile forms.

As there was no link between goals, metrics, data sources and adoption criteria, the data needed for reporting was ambiguous and poorly structured in RPP1.

4.2.2 **Implement**

RPP1 was implemented as a custom web application using the J2EE application framework. As reports were not clearly defined, developers focused on data collection (forms configuration). The main objective was to include all data items clinicians considered relevant for tracking clinical problems seen by the resident during practice. This resulted in long forms and long hard-coded lists of diagnoses. Data entry was cumbersome and the clinical experts who reviewed the RPP prototype “got lost” trying to navigate the application. Furthermore, as the development of the prototype followed a custom web application approach, forms and reports were hard-coded with no mapping to

the database. This fact made the configuration of forms and reports in RPP1 difficult and hard to maintain (e.g. it was hard to keep consistency in the terms used between data collected and reports).

RPP1 database was generated using an Object Relational Mapping (ORM) tool called hibernate with no support for multi-dimensional attributes. ETL was required to structure data for reporting, which made impossible to generate reports in real-time. The disconnect between clinicians and the development team was evidenced in the poor configuration of the forms and reports in RPP1.

4.2.3 **Evaluate**

Technical experts demonstrated RPP1 to clinical experts. The prototype was not well received therefore there was not even an attempt to conduct a Think Aloud session with residents to evaluate the application. The feedback obtained from clinical experts was used to model RPP2 in a more systematic way.

4.2.4 **Results**

During the initial version of RPP, the engineering research team thought they were following an agile BI approach (software development focused on information needs). However, from our review and understanding of the initial iteration of development, researchers were following instead a custom web development approach. RPP forms and reports were hard coded and the database was poorly structured for performance monitoring reports (relational database). As a result, data entry was cumbersome and it was difficult to support multi-dimensional, real-time reports. The

resulting RPP was not well received and revealed a great deal of disconnect between clinicians and the development team.

4.3. RPP 2

After failure of the initial version of RPP, development efforts were redirected towards improved development architecture and a more systematic way to model the healthcare monitoring application, configuration of the reporting database, and configuration of forms and reports linked to the reporting database. Also, there was a more systematic approach to user experience walkthroughs (using formal think-aloud sessions). A retrospective analysis of evidence collected during the development of RPP2 (review of application screenshots, analysis of database models and, analysis of think aloud sessions and expert checklists) was done to confirm that the development of the application was successful because of all the aspects of our methodology.

4.3.1 **Model**

During the development of RPP2, monitoring requirements were more precise and included the analysis of resident's clinical experience from multiple dimensions (e.g. own visits compared to global average; visits with more reading needed, more cases, comfortable; visits by time frame, visits by location). Also, there was a need to streamline the data collection process in way that data entry was not cumbersome but data collected was still useful to generate meaningful reports. This required a more systematic way of modeling metrics linked to goals and to data sources. The definition of metrics and data collection needs was mediated by the adoption criteria established by the clinical and

technical experts. Three iterations of development (RPP2a, RPP2b, RPP2c) were followed to come to the major design breakthroughs needed to address adoption criteria.

At a high level the goals monitored in RPP2 included:

- Reassure the resident there are not significant gaps in his/her clinical training experience: curriculum coverage by demographics and diagnosis (including procedures).
- Detect and re-mediate gaps in resident's clinical training: self-assessments.

The following key elements were defined as adoption criteria for RPP2:

- Residents should log their clinical experience (patient visits) in less than 30 seconds on a mobile device. This criterion evolved from an easy to use application in RPP2a to a more specific, measurable adoption criteria of logging visits in less than 30 seconds for RPP2b and RPP2c.
- Residents should see reports that map clinical experience in relation to the family medicine curriculum. For RPP2a, the adoption criteria were to track the exposure to clinical problems with centralized reporting. There was no clear mapping of the family medicine curriculum in RPP2a, resulting in poor feedback from reports. As a result of the evaluation phase in RPP2a, this adoption criterion was refined to tracking exposure to clinical problems mapped to the family medicine curriculum with customizable filters and in-depth analysis options in RPP2c.

Metrics monitored in RPP2 include:

- # Visits by day
- # Visits by age/gender
- # Visits by Clinical Domain (all)
- # Visits by Clinical Domain/Diagnoses
- # Notes by day

During the third iteration of development of RPP2, more refined performance metrics were defined that allowed residents to compare their own practice against average of peers. Measures added during this iteration include:

- Average # of visits per resident
- Average visits per diagnosis

4.3.2 **Implement**

The research team realized that, in order to effectively monitor goals in RPP2 and meet the adoption criteria, the database structure used for the initial version of RPP needed to change from an ORM (transactional database) to a multi dimensional star-schema model (optimized for reporting) in order to support a more sophisticated analysis of metrics (multi-dimensional analysis). Also, researchers realized that the amount of data collected needed to be streamlined. The data entry for one visit was cumbersome and it was hard for residents to remember what information they had already entered. This fact led to a careful analysis of the family medicine curriculum in order to define the key

dimensions needed for reporting and determine the right values to populate form controls and report labels.

The database model evolved from a star-schema model with no mapping to the family medicine curriculum in RPP2a, to a star-schema with mapping of clinical dimensions to dimensional tables in RPP2b. Changes in the database model allowed multi-select of diagnoses, consistency between configuration of forms and reports (standardized data field values), and easy configuration of filters. This was achieved by linking each dimension table to a control in a form. The associated report was also linked to the same dimension table in the database. Therefore, values in dropdown fields and associated report labels were the same. For example, the dropdown field "Age" was linked to the dimension table "Age" that supplied the values for the dropdown "Age". The associated report "Age/Gender" is linked to the same dimensional table "Age" that populates the values for age labels in the report (Figure 7).

Although changes in the database model supported new reporting requirements and consistency between data collection and reporting, the RPP2 star-schema model had no support for dimensional hierarchy columns, which means each clinical domain was mapped to separate dimension tables. Figure 8 depicts mapping of clinical domains (adult, children and adolescent –coca-, maternity, elderly, end-of-life, procedures) to dimensional tables in RPP2.

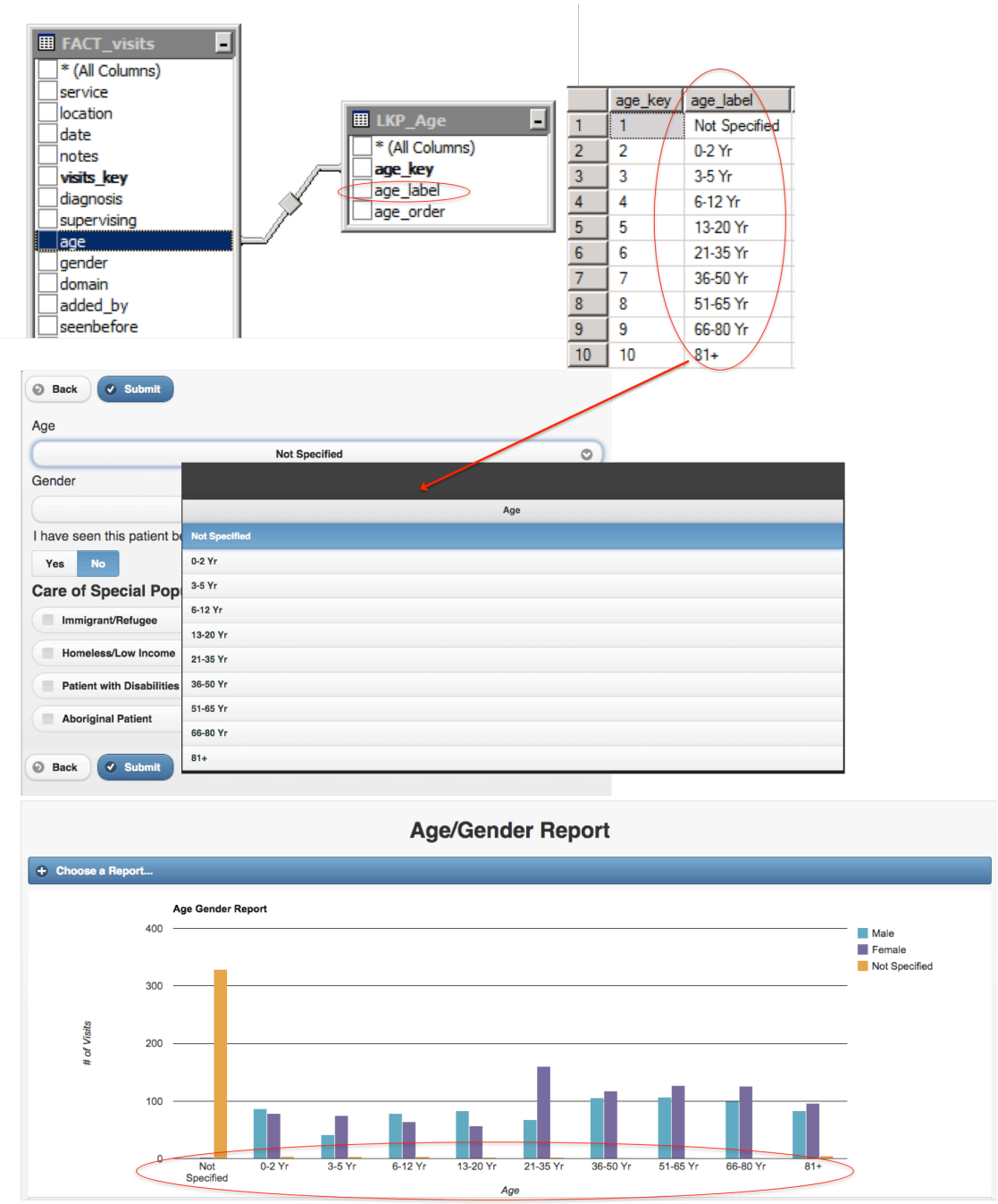

Figure 7: RPP Reporting Database to Forms to Reports. Example

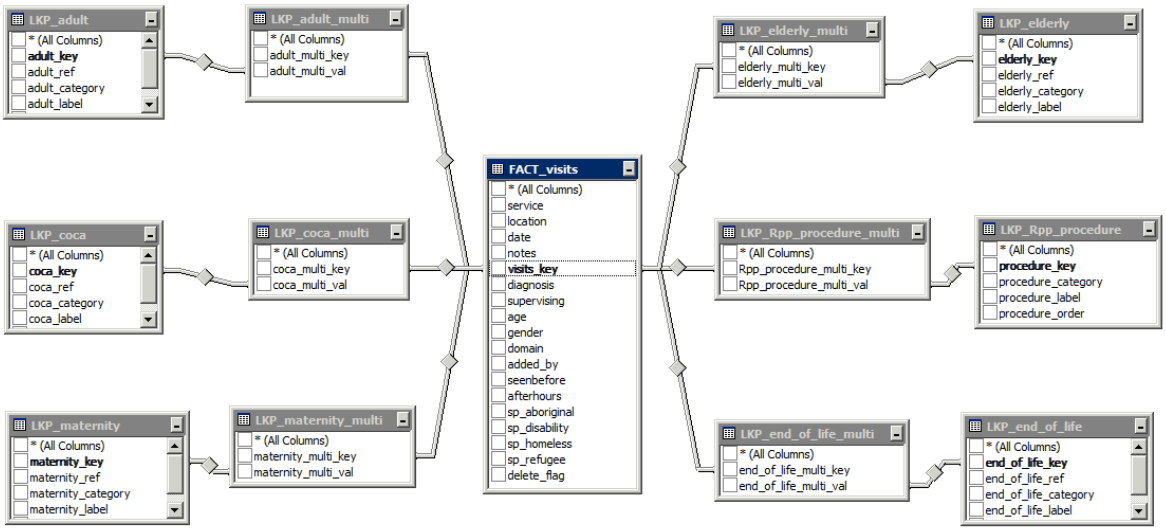

Figure 8: RPP2 Clinical Domains to Dimension Tables

RPP2 required complex JavaScript code to configure the UI. The fact that the RPP2 database model had no support for dimensional hierarchy columns led to a high level of effort on developers for the creation of reports. Although, multi-select was possible within one clinical domain, there was no support for diagnosis multi-select across clinical domains on a single visit (e.g. adults and procedures). For example, to generate a report as the one depicted in Figure 9, multiple queries in the backend had to be written as well as complex JavaScript for the UI.

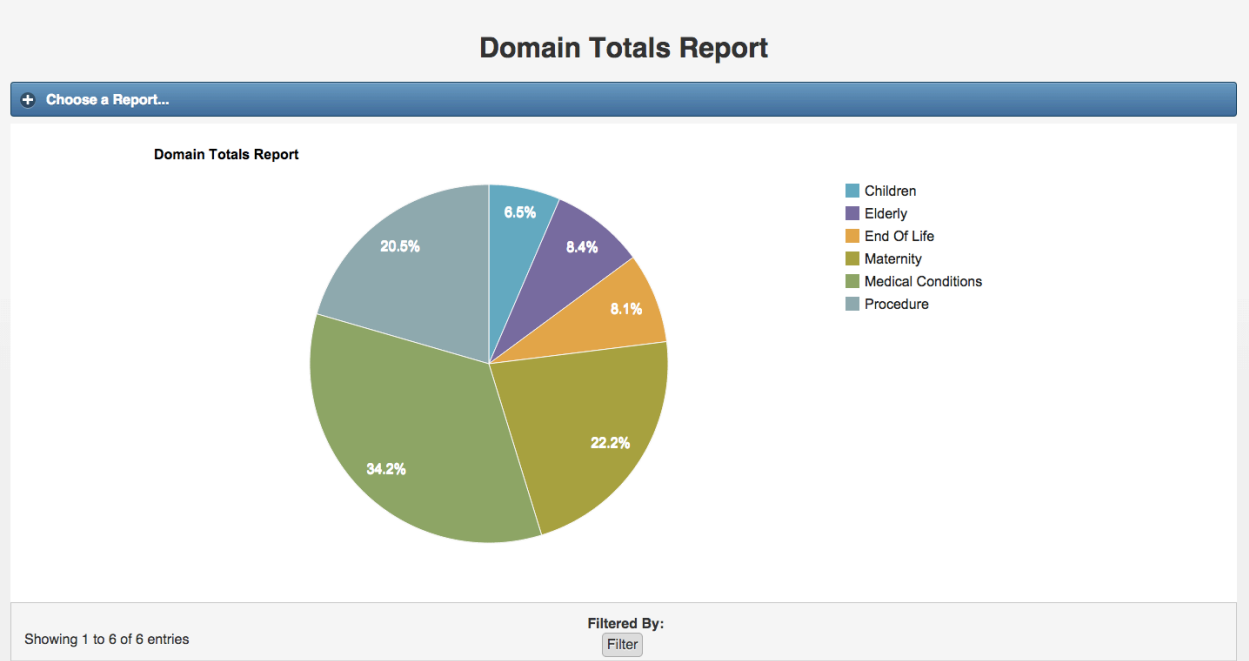

Figure 9: RPP2 Domain Totals Report

4.3.3 **Evaluate**

Three Think Aloud (TA) sessions were conducted during the development of RPP under the supervision of Doug Archibald, the education scientist researcher on the team. Complete transcripts were made and systematically evaluated by myself under the supervision of Doug Archibald. During development, the TA sessions were used to evaluate progress, gather feedback and identify key features for adoption and usability. The first session took place in November 2012, and the remaining two sessions were conducted in June 2013. Participants included members of the research team (i.e. health scientist researcher, technical and clinical experts, developers) and end users (i.e. residents). Additional to these sessions, an informal Think Aloud session (not recorded) was conducted on March 2013. Table 2 details TA session duration, dates, participants and roles.

Table 2: RPP2 Think Aloud Sessions

TA Session	Duration	Date	Participants	Role
1	51:12	November 2012	1 Scientist Researcher	Session Leader
			1 Clinical domain expert 1 technical expert 3 developers	Observers
			1 Resident	End user (tester)
2	Not Timed/ Not Recorded	March 2013	1 Scientist Researcher	Session Leader
			3 Physicians 1 Technical expert 1 Developer	Observers
			3 Residents	End user (tester)
3	58:50	June 2013	1 Scientist Researcher	Session Leader
			2 Physicians 1 Technical expert 2 Developers	Observers
			2 Residents	End users (testers)
	01:02:05	June 2013	2 Physicians 1 Technical expert 2 Developers	Observers
			3 Residents	End user (testers)

At the beginning of each session residents were introduced to the new version of RPP and were asked to log visits, as they would do during a regular patient encounter. Residents were reminded to talk out loud what they were doing while using the RPP app. Developers and domain experts took notes on users' verbalizations. New requirements (new app features or enhancements) emerged from these sessions and technical and clinical experts used the results to generate checklists (Table 3) that mediated changes in the model and implementation phases during the next iteration of development. Checklists were also used to quantify the extent by which the development of the application was meeting the adoption criteria.

Table 3: RPP2 Expert Checklists

Adoption Criteria	**Checklists RPP2b (April 3, 2013)**	**Checklists RPP 2b(May 29, 2013)**	**Checklists RPP2b (June 17, 2013)**
Time to log a visit <= 30 seconds on a mobile device. Visits for a whole day <= 10 minutes.	− Group data entry (pop-ups) by tracking, demographics, assessment, and self-assessment. − Pre-populate tracking data (service, location, supervising physician and date) − Diagnosis organized by domains/categories − Generate a 'Visits' list with option to filter by: user, date, block − Generate Notes lists with option to filter notes − Use radio buttons for Self-Assessment (reading needed; more cases needed; confident in my approach) − Include value 'Not Specified' in all dropdown boxes − Use truncated diagnosis labels for quick lookup − Restrict data entry to item selection for quick entry (dropdowns and checkboxes)	− Only Admin can select/delete a visit − Have option to filter visits − New Visit button at the bottom of the screen where number of visits is greater than 15 visits − Default tracking data to previous entry with each login − Age range defaults to corresponding domain in assessments: Children, adult, elderly, maternity, end of life, procedures − Quick access (tabs) for visits	− Use of radio buttons for demographics data entry (age/gender) − Display diagnosis labels in the visits summary
RPP reports should map resident's clinical experience in relationship to the Family Medicine curriculum on a near real-time basis.	− Separate report for procedures (similar to diagnosis report) − Use table reports in all reports. − Allow user to sort records by table columns. − Allow users to filter visits and notes. − Allow users to enter notes by patient, procedure and diagnosis.	− Create reports: Total # visits; average number of visits per resident; total visits/day − Report on total visits by resident by: (1) age/gender report; (2) special populations report; (3) Domains	− Allow users to filter visits and notes by self-assessments (comfortable, more cases, reading) and/or diagnosis − Report average, min/max number of visits/resident (same cohort) − Report resident's own visits vs. aggregated

For example, in Table 3 we see that one of the items in the checklist was to group data entry by main clinical dimensions: tracking, demographics, assessment and self-assessments. This re-organization of the form affected the time required to complete a

record. Data entry was faster, having the resident only to access those dimensions where data needed to be added or edited. Other items in the checklist included: 1) tracking information should be pre-populated from previous entry and 2) use of summary controls that allowed residents to see at a glance information entered for each of the clinical dimensions. All these items contributed to achieve the time benchmark to log a visit in less than 30 seconds. Also, it was possible to fit all data for a visit in one screen and there was no need for residents to scroll up and down to see whatever data they had recorded for the visit and "get lost" in the application.

The items in the checklist previously discussed came from one of the Think Aloud sessions where the resident expressed:

> *"I: The first thing I notice is that it would be useful to have an auto date fill in.*
> *D: You want it to default to todays date?*
> *I: Probably, because for the residents I know every second counts. You could save two clicks."*

Another segment in the Think Aloud session that gave insights on this respect was:

> *"I: The two keys things will be length or time of use. And the second one would be easy feedback so that the residents get some information. Getting a nice report that people will see I think will be nice to fill out."*
> *...*
> *"D: And your first impression of the user interface?*
> *I: Yeah I could use that. But like I said some of the drop downs and menus are not ideal. At least in terms of content so far."*

4.3.4 **Results**

The retrospective analysis of RPP2 uncovered important mechanisms that led to a successful delivery of RPP2. First, as RPP evolved from a simple data collection tool to a more sophisticated analysis tool, a better configuration of the database was needed. In RPP2 the database configuration was linked to performance reporting needs (metrics). Metrics were linked to goals and adoption criteria and, reporting needs combined with adoption criteria were used to streamline the collection of data to compute metrics. Particularly, mapping of clinical domains in the family medicine curriculum to the reporting database was key to support multi-select of diagnoses in forms and multi-dimensional analysis of visits. The configuration of forms and reports done by linking the form controls and its associated reports to the appropriate dimensional tables in the database resulted in increased consistency between data collected and metrics reported.

Another important mechanism that affected the effectiveness of RPP2 development was the use of Think Aloud method combined with expert checklists. The scientist researcher and the student researcher for this thesis did a formal analysis of the TA sessions. Each session was transcribed "verbatim" and we met twice to analyze transcripts. During the first meeting, the scientist researcher guided the coding process. The first transcript was divided into segments; codes were assigned to each segment, and then grouped into subthemes and main emerging themes (Table 4). For the analysis of the remaining transcripts we took a similar approach, however, the coding of transcripts was done individually. The main emerging themes uncovered during our analysis include: 1) enhancements, 2) features for adoption/usability, and 3) security. Table 4 depicts an example of the analysis of the first TA.

Table 4: RPP2 Think Aloud 1. Emerging Themes.

Emerging Themes	Sub Themes	Codes	THINK ALOUD Transcript-Segment
Enhancements	Format / Layout	Reports Time	I: I see "choose a report" so **if there was "choose a time frame"**. Below the same bar that you would have weekly, monthly and then total.
	Personal Preferences	Self-Assessment	I: Yeah we tend to be picky on words but I think none of the residents will put "mastered" because you are not supposed to. I think it should be something more like **"comfortable" vs. "not comfortable"**. So if have it said "comfortable seeing without supervisor." Or maybe **just a scale of comfort**. Comfortable-completely lost. Or something like that and then you would **have at least a more intuitive scale to put things into**. Because more visits needed and need to read might be the same
Features for Adoption Usability	Curriculum	Location	I: I'm **not sure what location purpose is**. Let's say they want to know if you TOH in the clinic or in EMRG or something like that, but I **would tend to see the services as more useful**. D: What is an example of a list of services? I: Family medicine, Emergency room... DD: Would these be the rotations or even more specific? I: Yes, the rotations that we do. D: You don't want location, or rotation you want that called service? I: Yes. D: And you would like that up at the top as the first or second one? I: Yep, because we do month by month
	Access to Data	Notes-Flag Notes	I: especially for a visit that you wrote a note if you can **access it separately through notes then you could flag it** so you **don't have to go through the whole visit**. That we will know once we start using it a bit more. D: If there is a note from a visit and a note file it would be nice if you could **click from there to the visit** A: Would that also mean that you could link a note to a visit? D: No if they want that they should just go to the visit and put the note in.
	Data Entry	Well vs. Sick	2: Do you have the **well vs. sick for all the patient groups**? 2: Probably, because ultimately you use these for the charts as opposed to the diagnosis code, which would be like hundreds and hundreds. **So it would make sense that for adults we would have some too**.
		Autofill	I: The first thing I notice is it would be useful to have an auto date fill in. D: You want it to default to todays date? I: Probably, because for the residents I know every second counts. You could save two clicks. I: And are these "location attending" are they going to stay the same as the previous one? D: Yes, so the first time you enter them or change them. Are you fine that they stay the same as what happened 12hrs ago? I:I think that is reasonable.
	Feedback	Reports	I: The two keys things will be length or time of use. And the second one would be **easy feedback** so that the residents get some information. **Getting a nice report** that people will see I think will be nice to fill out D: Do you have idea of report you want to see that is not there now? I: Once you have the **list of procedures** that would be an important report to add and then some of the **paediatric domain reports** as well
	Performance/Responsiveness	Drop downs configuration	D: And your first impression of the user interface? I: Yeah I could use that. But like I said **some of the drop downs and menus are not ideal**. At least in terms of content so far Drop downs: Procedures, newborn and elderly
Security	Security options	Password security	I: My concern is not people playing tricks on each other but inadvertently coming to a computer and **accidentally entering it on someone else's account**. Because we do share stations at the clinic

Results from the analysis of TAs were used to evaluate how useful this method was for the improvement of RPP by confirming emerging themes aligned to the progression of RPP screenshots.

Figure 10 depicts an example of RPP progression based on feedback obtained from one of the Think Aloud sessions. The resident expressed he liked the use of tabs (See Table 4-Data Entry) for data entry, as the diagnoses could be hundreds. We see how this change was implemented in RPP2.

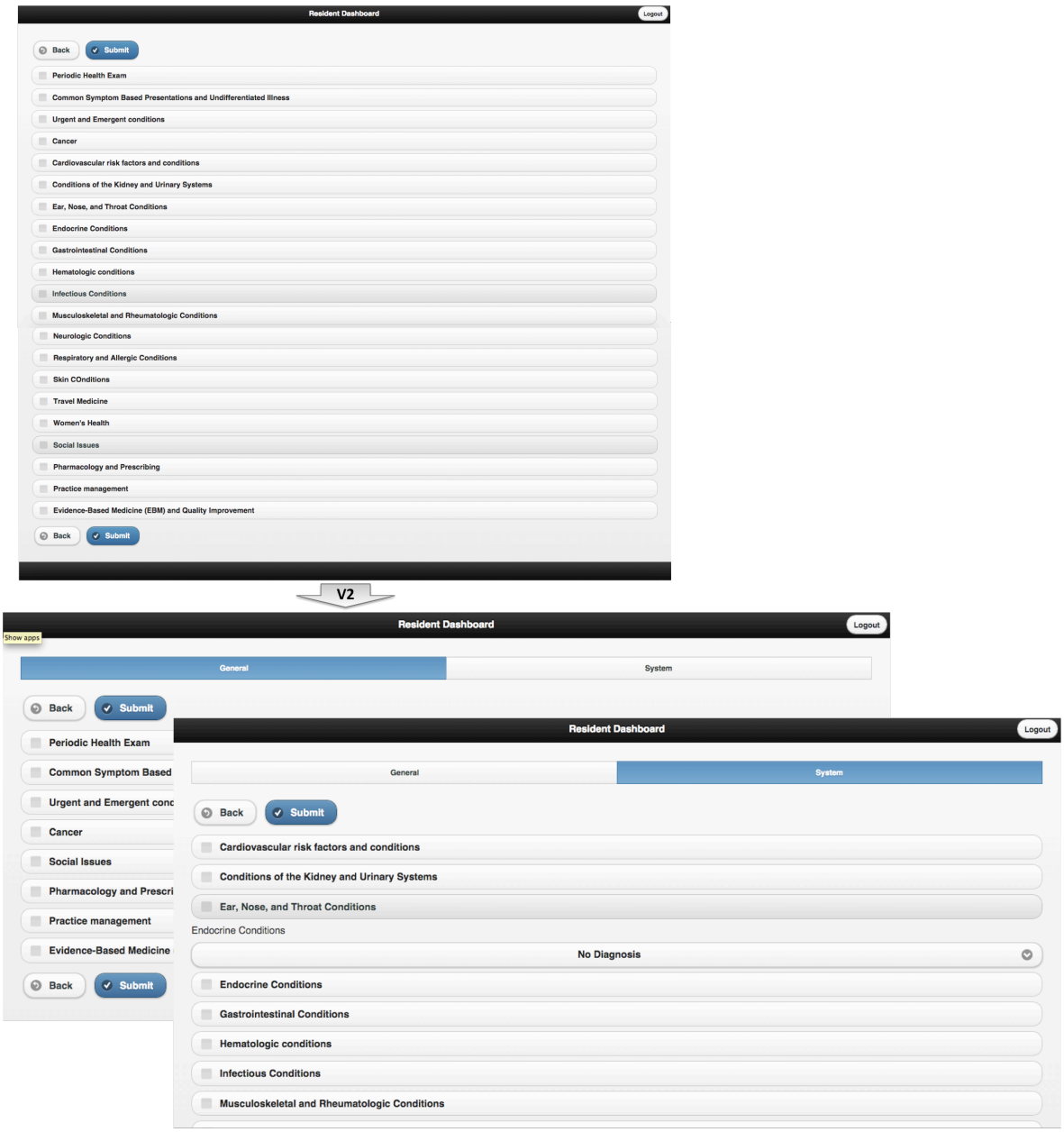

Figure 10: RPP Think Aloud Themes to Screenshots –Progression V1->V2

4.4. RPP 3

4.4.1 **Model**

RPP3 was developed using our development methodology. During this iteration, the development team and clinicians were in synch and only minor refinements to RPP were required in terms of reporting (metrics). No major changes were made to the UI for data collection, however, significant improvements to the application architecture allowed developers to reuse controls for a faster configuration of forms and reports (Mata et al., 2015). RPP3 goals remained the same as in RPP2. As for the adoption criteria, experts determined that:

- Residents should be able to analyze data from multiple dimensions (filters and drill through capabilities)
- Doctors responsible for family medicine should be able to configure the system through a web browser.
- Third-party reporting tools must be supported for in-depth analysis to improve family medicine program.

A total of six new/refined reports (metrics) were defined for this iteration of development:

- # Visits by self-assessments
- # Missing Assessments
- # Visits by Location
- # Visits by Block/RPP Week (calendar control)

- min/max number of visits (for each diagnosis and each category of diagnosis -two level hierarchy-)
- Assessments by category as a percentage of total assessments.

4.4.2 **Implement**

During this iteration of development, the focus was mainly on reporting capabilities (i.e. support third-party reporting tools and drill-through capabilities). Also, a refinement of the application architecture was required to reduce the level of effort on developers to configure forms and reports. Another key aspect addressed during RPP3 implementation was how to provide clinical administrators with the ability to configure any of the clinical dimensions values for data collection.

In order to support third party reporting tools and drill-through capabilities, the database model was refined one final time. We determined that 'Assessments' was one of the key clinical dimensions that need to be reconfigured in the database model. In RPP2, each of the six clinical domains was considered as a separate dimension. In this iteration of development, we mapped assessments as one single clinical dimension to the database. All six tables corresponding to clinical domains (i.e. adult, children and adolescent, maternity, elderly, end-of-life, procedures) were consolidated into one single assessment table that included clinical domains and category as hierarchy columns (Figure 11). Therefore, reports using third party tools (e.g. MS Excel Pivot Tables) were supported as well as drill-through capabilities as shown in Figure 12.

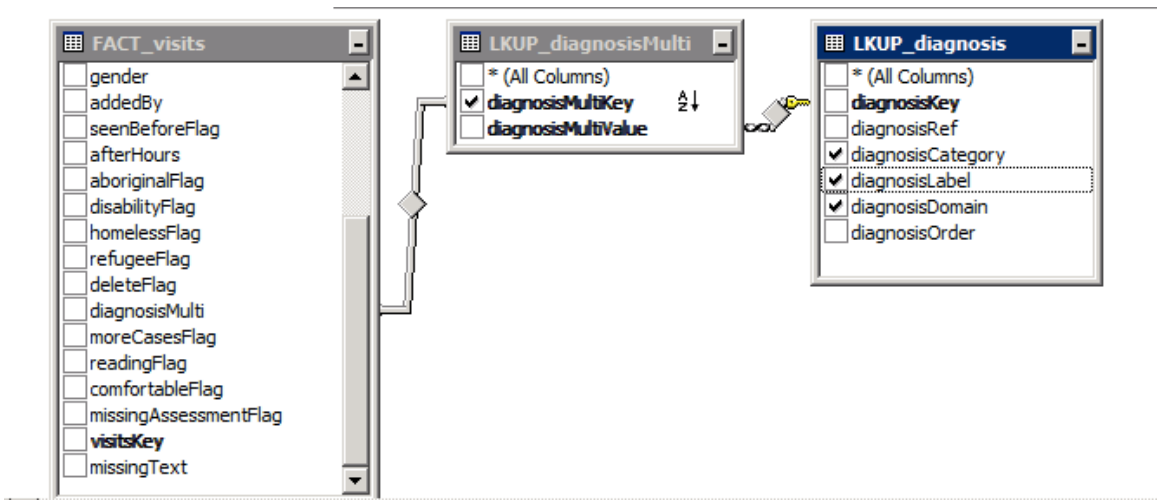

Figure 11: RPP3 Database Model – Assessments

	A	B	C	D	E	F	G	H	I	J	K	L
1	Diagnosis	Username										
2		alabelle									alabelle Total	Grand Total
3	Clinical Domains	0-2 Yr	13-20 Yr	21-35 Yr	3-5 Yr	36-50 Yr	51-65 Yr	6-12 Yr	66-80 Yr	81+		
4	⊞Adult		12	125		201	213	3	181	92	827	827
5	⊟Coca	124	92	9	62	6	9	76	8	1	387	387
6	⊞Cardiac concerns	2			1						3	3
7	⊞Childhood Abuse and Neglect											
8	⊞Endocrine conditions											
9	⊞Eye conditions	1			1	1					3	3
10	⊞Gastrointestinal concerns	8	2		2			2			14	14
11	⊟General	79	34		21			33			167	167
12	Adolescent health prevention		18					3			21	21
13	Allergy and Anaphylaxis		1					3			4	4
14	Eighteen month pediatric visit including autism spectrum screening	6									6	6
15	Fever (undiagnosed)	4			1						5	5
16	No Diagnosis											
17	Routine Well Infant or Child Assessment	67	14		20			27			128	128
18	Unwell child (no diagnosis)	2	1								3	3
19	⊞Genetic/developmental conditions	1	9	1	5	1		18			35	35
20	⊞Injury Prevention											
21	⊞Mental health issues		21					1			22	22
22	⊞Muskuloskeletal conditions	1	3		4			6	1		15	15
23	⊞Neurologic conditions	1	2		1		1				5	5
24	⊞Pediatric emergencies	3									3	3
25	⊞Renal, urologic, genitourinary issues	4	3		3		2	1			13	13
26	⊞Respiratory conditions	20	9	8	23	4	6	15	7	1	93	93
27	⊞Sexuality, contraception issues		3								3	3
28	⊞Skin conditions	4	6		1						11	11
29	⊞Elderly						4		50	63	117	117
30	⊞End Of Life					2	15		41	11	69	69
31	⊞Maternity	2	2	35		9					48	48
32	⊞Procedure	2	1	6		5	8		7	4	33	33
33	Grand Total	128	107	175	62	223	249	79	287	171	1481	1481

Figure 12: RPP3 Drill through Reporting – MS Excel Pivot Tables

Moreover, the consolidation of clinical domains into one single dimension allowed residents select multiple diagnoses across different domains in one single visit. For example, the visit summary shown in Figure 13 depicts the log for a visit associated to a patient who presented with a fracture (diagnosis under medical conditions) and a cast was applied (procedure) during the same visit. The visit includes the selection of multiple diagnoses across two different domains of care, two diagnoses from Medical Conditions and one from Procedures.

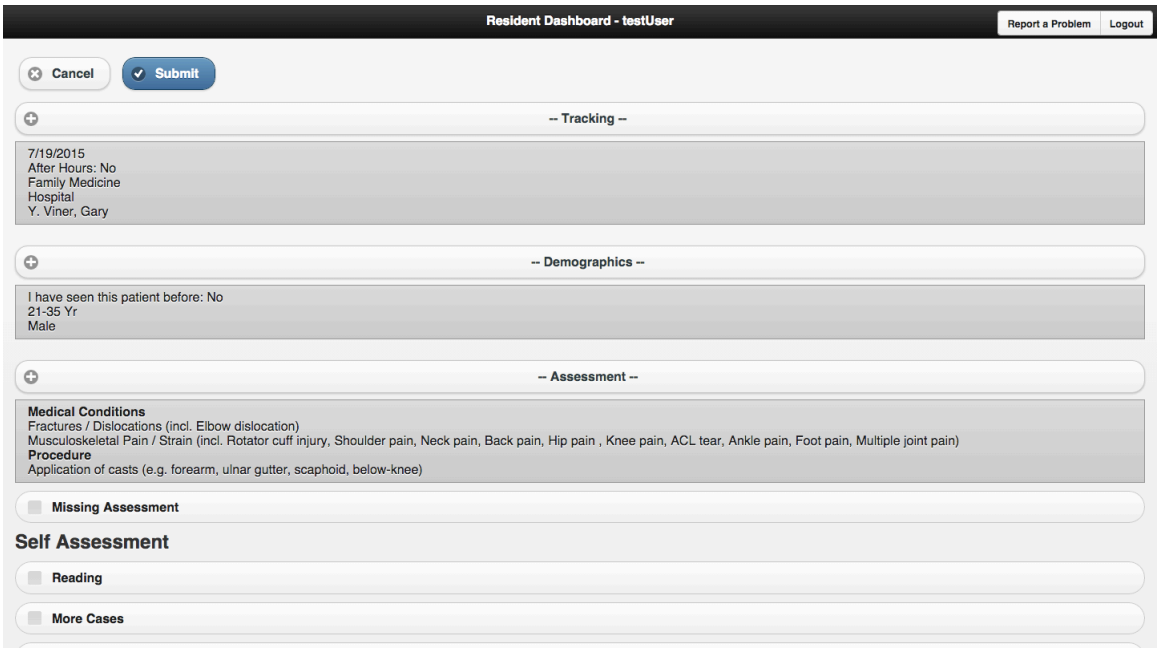

Figure 13: RPP3 Multi-Select across Clinical Domains

4.4.3 **Evaluate**

During this iteration of development, the evaluation of innovations in RPP was done via ad-hoc user experience walk-throughs and meetings with the domain experts and the development team. There was no need to conduct additional think aloud sessions,

as the thought process of end-users in terms of usability was clearly understood during think aloud sessions in RPP2, and the same end-users were providing feedback in RPP3. The main feedback gathered during this iteration of development was on the refinements of reports, which led to refinement of metrics in the performance model. The feedback was complemented with interviews done by D. Archibald, the education researcher and analyzed by myself. The major innovations that resulted from the evaluation during this iteration of development included: improved performance benchmarks (min/max visits by category); grouping of medical conditions common to all clinical domains into one single clinical domain; calendar control to track visit logs. These new requirements on metrics led the development team to refine the database configuration, which reduced the engineering effort and enabled the support of third party reporting tools and drill trough reporting capabilities with ease compared to RPP2.

The interviews were transcribed “verbatim” and analyzed by myself. Figure 14 depicts the emerging themes from the analysis of interviews and confirmed that, features for adoption and usefulness are key factors to consider for the development of a CPMA for performance monitoring of residents.

Count of Themes (Parent Notes)	
Row Labels	**Total**
Accessibility	7
Features for adoption	**21**
Perceived Risks	**11**
Usability	1
Usefulness	**23**
(blank)	
Usefulness;Perceived risks	1
Usability;Features for adoption	2
Features for adoption; Usefulness	1
Usefulness; Features for adoption	1
Features for adoption;Usefulness	1
Features for adoption; Intention to use	2
Features for adoption; Data Quality	1
Features for adoption; Perceived risks	1
Perceived Risks; Data Quality	1
Usefulness; Data Quality	1
Connectivity;Accesibility	1
Usability; Features for adoption	1
Grand Total	**77**

Figure 14: RPP Interview Analysis

4.4.4 **Results**

The key fact from this iteration was the right mapping of assessments to the reporting database. Assessments, as defined by our development methodology, are grouped into one clinical dimension having the domain, category and diagnosis as a hierarchy columns rather than separate dimension tables. If two or more clinical dimensions (i.e. clinical domain) can take more than one value for a visit and can be grouped in nested hierarchies (i.e. domain, category, diagnosis) then they should be defined as a hierarchy column into one single dimension table. This enables drill-through

reporting and makes the database readily available for use with third party reporting tools. Also, the level of effort required by either, the developers or clinical administrator to update values in the dimension table is reduced from many (i.e. 6 clinical domains) to one. Another highlight from this iteration of development was the dramatic reduction in the number and complexity of queries required to report metrics. In RPP2, 59 queries were written versus 26 queries in RPP3. This was mainly due to a better mapping of clinical dimensions (i.e. assessments) to the reporting database.

4.5. Summary

After we reviewed the literature and understood what experts and the development team said they were doing, we defined the first version of our development methodology that captured the steps a methodology for the development of CPMA should include.

A careful review of RPP2 was done to confirm the steps outlined in our development methodology were helpful to explain the mechanisms that led to a successful deployment of RPP2. Of particular interest is the fact that Think Aloud sessions by itself were not enough to meet usability and user adoption criteria, and expert's checklists were needed to effectively translate user's feedback into actionable development efforts that met adoption criteria. Another key aspect captured in our development methodology, and confirmed with our case study, was the right configuration of the database, i.e. mapping of clinical dimensions –tracking, demographics and assessments- with support to multi-hierarchy columns. This reduced the effort on developers and allowed the development team to deliver faster and more

efficient application features that met adoption criteria. Also, the right mapping of clinical dimensions to the reporting database made the configuration of dimension values easier by either developers or clinical administrators.

Chapter 5. Evaluation

In this chapter we evaluate our thesis based on the results of our case study with RPP. In section 5.1 we evaluate how well our proposed methodology was confirmed by our analysis of the evolution of RPP from RPP1 to RPP3, especially with the use of thematic analysis of the think aloud session transcripts for RPP2a, RPP2b, and RPP2c. In section 5.2 we evaluate our proposed methodology by comparing it to the related works identified in section 2.4, using the evaluation criteria that we identified in section 3.3. Finally, in section 5.3 we compare the application that was created using our methodology, RPP3, with other similar applications that were built using custom web development.

5.1. Evolution of RPP

The purpose of this section is to evaluate our development methodology across the three iterations of development of RPP. Particularly, we will review how our methodology is useful to explain how better results were obtained at each iteration of development and the development aspects that did not work well. We trace through tables, the key factors in our development methodology that helps explain the delivery of a more robust and usable RPP application.

In Table 5 we review the engineering effort required at each iteration of development in RPP.

Table 5: RPP Evaluation - Engineering Effort

Engineering Effort	**Prototype (J2EE)**	**RPP 2a (QuickForms 1.0)**	**RPP 2b (QuickForms 1.0)**	**RPP 2c (QuickForms 1.0)**	**RPP 3.0 (QuickForms 3.0)**
Form Factors	PC browser optimized	All form factors	All form factors	All form factors	All form factors
Form/Report Linkage	Hard	Easy	Easy	Easy	Easy
Third-party reporting tools	Hard-coded reports	Simple ETL (flawed star schema)	Simple ETL (flawed star schema)	Simple ETL (flawed star schema)	Multi-dimension hierarchies
Application Configuration	UI and reports hard coded	Moderate JavaScript for business logic	Moderate JavaScript for business logic	Moderate JavaScript for business logic	Easy Packaging of pre-defined components
Repeatable (Defined Method)	Hard coded forms and reports	Moderate Dimensions mapped to dropdown values Custom layout code	Moderate Dimensions mapped to dropdown values Custom layout code	Moderate Dimensions mapped to dropdown values Custom layout code	Easy Dimensions mapped to dropdown values Library of controls and templates used to assemble layout
User-Centered Design	No. Major disconnects	Yes	Yes	Yes	Yes

RPP1 was developed using a J2EE framework. The user interface was a browser based HTML interface that worked well in a PC but did not support other form factors. The three tiers of the application in RPP1 were hard coded, which increased the level of effort required by developers. Any change in RPP1 forms, generated a new ORM database, thus the report associated to the form had to be also adjusted in order to keep consistency in the app. Moreover, the ORM generated database in RPP1 was not a multi-dimensional database, thus third party reporting tools were not supported.

In RPP2, the mapping of clinical dimensions to a star-schema database resulted in a moderate level of effort required by the development team to configure the application. The linkage of forms and reports was fully supported by the new database configuration.

The dropdown values in a form were linked to a dimensional table and the same dimensional table was used to populate the report associated to the form. Also, the fact that clinical dimensions were mapped to dimensional tables in the database facilitated the reorganization of forms by grouping clinical dimensions in tabbed controls, which made data entry as well as data display more efficient (all form factors were supported). However, RPP2 business logic was programmed in JavaScript and the forms and reports layout was custom coded, which required a moderate level of effort by the development team. Also, as multi-dimension hierarchies were not supported, some simple ETL was needed to prepare the data for reporting metrics.

In RPP3, the clinical dimensions were clearly identified and mapped to different dimensional tables. The six clinical domains (mapped to separated tables in RPP2) were merged into one single dimensional table with multi-dimensional hierarchies (domain/category/diagnosis). This change allowed an improved architecture framework that packaged pre-defined controls and templates in an application library, which made the configuration of RPP a matter of selecting controls and defining the corresponding dimension table to populate the values in the control. The result was a very significant (order of magnitude) reduction in the implementation effort.

Table 6 we review the application features that resulted at each iteration of development in RPP2 and that can be explained by the methodological steps in our development methodology.

Table 6: RPP Evaluation - Application Features

Application Features	**Prototype (J2EE)**	**RPP 2a (QuickForms 1.0)**	**RPP 2b (QuickForms 1.0)**	**RPP 2c (QuickForms 1.0)**	**RPP 3.0 (QuickForms 3.0)**
Data Entry	Complex form, long list and search for diagnosis.	Complex form: up to 2 min, but dropdown selection	30 seconds	30 seconds	30 seconds + Calendar control
Diagnoses selection	Search for diagnoses	Table to diagnoses control but inconsistent, long, awkward list	Table to diagnosis control; diagnoses hierarchies	Consistent simple navigation by age, and specialty	Medical conditions common to all, then navigate by age, specialty
Data Display	Long lists; long sentences	Long lists Long sentences	Templated summaries	Templated summaries	Templated summaries; Min_max reports
Drill-Through Reporting	No	No	Hard coded filters	Customizable filters (simple ETL)	Customizable filters
Near Real-Time Reporting	No. ETL required.	Yes	Yes	Yes	Yes
Security	Basic Login and secure hosting	Basic Login and secure hosting	Basic Login and secure hosting	Basic Login and secure hosting	Basic Login and secure hosting *Reviewed and reaffirmed

In RPP1, forms and reports were hard coded in complex forms with long lists of diagnosis. To select a diagnosis, the resident had to type in part of the diagnosis, which often resulted in a diagnosis not found, as it required the exact match of words and the resident had to remember the diagnosis description used for a particular condition. This made the data entry cumbersome, and frustrating for the end-user. The ORM generated database (relational database) in RPP1 was not optimized for reporting and ETL was required. Only simple reports were supported.

In RPP2a, the reconfiguration of the database from a relational database to a star-schema model (Table 7) enabled the generation of reports in real time. However, as the clinical dimensions in the family medicine curriculum were not mapped to the database, the data organization was not optimal, which did not represent a significant improvement compared to diagnosis selection in RPP1. The diagnosis selection list was still inconsistent, awkward and long. This suboptimal diagnosis selection list translated into about 2 minutes required by a resident to complete a visit. In RPP2b, the mapping of clinical domains in the family medicine curriculum to the reporting database (Table 7) was key for the reorganization of forms and reports. RPP forms were organized in sub-dialogues (clinical domains, categories, and diagnosis) and residents were able to select more than one diagnosis for a visit within the same clinical domain. To select a diagnosis, residents choose the domain of care, the category and finally the diagnosis (e.g. Adults/System/Endocrine Conditions/<Diagnosis>). This reorganization of forms (driven by the optimized data model) dramatically improved the time to complete a visit, from up to 2 min in RPP2a to 30 seconds or less in RPP2b and RPP2c. The drillable reports and customizable filters were enabled in RPP2c due to the mapping of clinical domains to the reporting database (Table 7). However, simple ETL was required to enable this feature.

In RPP3, the major innovation in terms of application features was driven by refinements in the performance model (refined metrics). The systematic evaluation of RPP via user experience walkthroughs with one of the resident and the clinicians, led the family medicine program directors to rethink the organization of the clinical domains. Diagnoses that were common across various clinical domains were grouped into one single clinical domain that applied to all, called Medical Conditions. This change was

possible given the new configuration of the database (merging of the six clinical domains into one single dimensional table –assessments-) that allowed residents to select diagnoses across domains of care (e.g. medical conditions>Allergic rhinitis, and children>routine neonatal assessment; parenting advice re: injury prevention (eg. car seats, helmets...). Also, ETL was not required for drill through reporting as the database was optimized for the analysis of multi-dimensional across the clinical domains.

In Table 7 we evaluate the development progress of RPP2, confirmed by our methodological approach, using the criteria set to evaluate a development methodology. One of the key factors that helped explain the major innovations between the iterations of development in RPP2, was the systematic evaluation of usability. Major disconnects between clinicians and the IT development team observed during the demo of RPP1 to clinicians were dramatically reduced by the use of usability testing methods centered on the user (i.e. user experience walkthroughs and think aloud sessions). These evaluation sessions led experts to refine the adoption criteria from general criteria (mobile accessible, centralized reporting and ease of use) to more specific, measurable criteria for adoption. In RPP 2b, this included time to complete a record in less than 30 seconds or, visits for the whole day in less than 10 minutes. In RPP2c, this included quick navigation of visits. And in RPP3 this included refinements to the reports. More importantly, the systematic evaluation of usability drove experts to refine the performance model. This included what metrics (reports) were needed and what data needed to be collected to generate those metrics. The most important improvement from RPP2a to RPP2b was the introduction of Think Aloud sessions in the application development cycle in order to get users' insights on usability of the prototype and elicit user feedback. The IT development

team got a more clear understanding on how clinical conditions could be organized more efficiently to improve data entry by residents and create forms that mapped to reports. This new reorganization of diagnoses into clinical domains and categories in RPP led to the creation of sub-dialogs and summaries that dramatically reduced the complexity and time required to log a new visit.

Table 7: RPP Evaluation - Development

Development Methodology	**Prototype (J2EE)**	**RPP 2a (QuickForms 1.0)**	**RPP 2b (QuickForms 1.0)**	**RPP 2c (QuickForms 1.0)**	**RPP 3.0 (QuickForms 3.0)**
Clinical performance (Goals, Metrics)	Track exposure to clinical problems	Track exposure to clinical problems	Multidimensional analysis Gaps in clinical mapped to FM curriculum	Performance benchmarks Support resident's self-assessment	(min/max number visit)
Clinical acceptance (Adoption Criteria)	Mobile accessible Centralized Reporting Ease of use	Mobile accessible Centralized Reporting Ease of use	Time to log a visit <= 30 seconds / whole day <= 10 minutes No Scroll	Quick navigation visits, self-assessments	6 extra reports
Model Driven (Reporting Database and Clinical Dimensions)	No BI reporting DB	Star-Schema No mapping of FM curriculum to reporting DB	Star-Schema FM curriculum to DB (Domains of care) No dimensional hierarchies.	Star-Schema FM curriculum to DB (Domains of care) No dimensional hierarchies.	Star-Schema FM curriculum to DB (clinical dimensions) Multi-dimensional hierarchies.
Integrate Forms, Reports and Dimensions	None	None	Dimensions mapped to forms and reports	Dimensions mapped to forms and reports	Dimensions mapped to forms and reports
Integrate Technical and Clinical Expertise	Major disconnect and hard to fix	Simple forms mapped to tables, mapped to reports	Sub-Dialogs Summary controls Multi-level selection	Drillable reports Customizable filters	Calendar control Min/max reports
Systematic Evaluation of Usability	Clinical and technical expert review	User experience Walkthrough (think aloud) Clinical and technical expert review	User experience Walkthrough Clinical and technical expert review	User experience Walkthrough (think aloud) Clinical and technical expert review	User experience Walkthrough Clinical and technical expert review

Key to these innovations was the mediation, in the form of clinical and technical expert's checklists, between issues raised during think aloud sessions and user experience walkthroughs and the IT development efforts. The checklists integrated clinical and technical expertise and merged adoption criteria, usability issues and functional requirements that guided the development team to come up with the innovations (simple forms mapped to tables in RPP2a; sub dialogs, summary controls and multilevel selection of diagnoses within a clinical domain in RPP2b; drillable reports and customizable filters in RPP2c).

The evolution of the performance model from simple metrics in RPP1 to more sophisticated metrics in RPP2b, RPP2c and RPP3.0, had a significant impact in the implementation phase. The major breakthroughs were achieved in RPP2b with the mapping of clinical dimensions to the reporting database. This optimization in the database model enabled integration of forms and reports. The second major breakthrough was achieved in RPP3.0, when the clinical domains were grouped into one single dimensional table with multi-dimensional hierarchies. This improvement on the mapping of clinical dimensions to the database enabled reporting of more sophisticated metrics with less effort on the development team (eg. 59 queries in RPP2c versus 26 queries in RPP3 to generate even more reports; clinical domain totals and min/max visits by clinical domain and category and calendar control to track residents' usage).

5.2. Related works

In Table 8 we evaluate our application development methodology in comparison with the three related works described in 2.4 using the three sets of criteria listed in 3.3:

engineering effort, application features and methodology. The use of EHR systems as data sources for performance monitoring or the use of a Custom Web Development approach are both approaches that require the highest engineering effort to develop a CPMA, such as RPP, as here there is no systematic methodology or specific architectural support. Agile BI has a systematic methodology but the architectural support is lacking. The key to reducing effort in our development approach is the direct linking of forms and reports through a database fully integrated with clinical dimensions optimized for both, data capture and reporting.

Table 8: Related Works –Engineering Effort

Engineering Effort	**EHR**	**Agile BI**	**Custom Web Development**	**Structured CPMA Methodology**
Form Factors	Not Supported	Not Supported	Depends on how it is coded.	Supported by QuickForms
Form/Report Linkage	Not Supported	Not Supported	Not Supported	Dropdowns values to report graph and charts
Third-party reporting tools	Not Supported	Supported	Possible but not optimized	Supported
Application Configuration	Propietary DB. ETL required.	Dimensional DB. ETL required	ORM. ETL required.	Clinical dimensions DB links forms and reports
Repeatable (Defined Method)	Hard	Systematic	Hard	Systematic + reusable controls and templates
User-Centered Design	Ad hoc	Supported	Ad hoc	Supported

EHR databases are proprietary relational databases, not optimized for reporting but patient care. Therefore, ETL is required for integrated reporting and, third-party reporting tools are not supported. On the data collection side, any new metric that requires mapping of a new data field to any EHR form is addressed by each EHR vendor, a process that could be slow and costly, requiring that each of the EHR vendors adjust their systems to include the new data fields required for performance monitoring.

Depending on the development methodology followed by each vendor, a user-centered design approach may be suitable for the development of integrated performance monitoring reports in EHRs. Moreover, interoperability issues across different EHR vendors add to the engineering effort needed.

Agile BI promotes collaboration among stakeholders during all phases of application development. BI apps are built using dimensional data models; therefore, third party reporting tools can be easily used. However, the main limitation of this approach in healthcare environments is the timely access to data sources, availability of the right data, and the need of ETL processes, which dramatically increases the engineering effort. Agile BI is a generic approach and there are no specific guidelines on how to configure clinical dimensions to the reporting database. However, once the data is available in the data mart, the configuration of reports is easy. On the application evaluation side, agile BI approach is an improvement upon EHR, in that the business user writes user stories used to test the application. However, the end-users do not interact with the technology during development, which limits the evaluation of the tool to a pass/fail of functions defined by the business user in the user story. Rich data that emerge from end-user interacting with the tool are missed.

A custom web development approach for RPP requires a high level of engineering effort. The ORM generated database is not optimized for performance monitoring and ETL is required, which is often a tedious task. There are two different databases, one used for data collection and one for reporting; therefore independent configuration of all the three application tiers is required, which increases the engineering effort to maintain

and develop new applications. Moreover, forms and reports are hard coded for each application, which makes this development approach hard to repeat. Also, it is hard to address usability issues in different iterations of development because of the rigid development approach. Changes in the three application tiers increase the engineering effort, which also affect the cost for a project.

The engineering effort required in the structured CPMA methodology is mainly affected by the reporting database configuration. Key to this approach, in terms of engineering effort, is the mapping of clinical dimensions to a star-schema database. Each form and its associated report are mapped to the same dimensional tables in the reporting database, which dramatically reduces the effort needed to configure the application. An additional column in the dimensional table represents an additional control in a form and an additional hierarchy in its associated report. Moreover, a more structured way of configuring the application based on a performance model (metrics to database dimensions to forms and reports) makes this methodology easy to repeat and compatible with user-centered design. Technical aspects of application development are optimized, which enables the IT development team to react faster to usability requirements. Moreover, the multi-dimensional database model is compatible with third-party reporting tools, so business users do not need to wait for the IT development team to build the cubes or data marts for in-depth analysis of data.

In

Table **9** we evaluate applications features that can be supported if RPP were developed using each of the four different development approaches.

Table 9: Related Works - Application Features

Application Features	EHR	Agile BI	Custom Web Development	Structured CPMA Methodology
Data Entry	NA	NA	Ad-Hoc	Optimized
Diagnoses selection	Inconsistent	Inconsistent	Ad-Hoc	Optimized
Data Display (Templated Summaries)	NA	NA	Ad-Hoc	Optimized
Drill-Through Reporting	Limited	Yes. Incomplete	No	Yes
Near Real-Time reporting	ETL	ETL	Yes	Yes

EHR falls short on all of application features for RPP. The diagnosis selection is inconsistent as it depends on the EHR configuration at each of the multiple locations where residents see patients. The diagnosis descriptions vary depending on EHR deployed at each location, thus the same clinical condition could be described differently in each EHR or even not be available. Moreover, EHR database schemas (relational database) are not optimized for reporting, thus ETL is required to support drill through reporting in integrated reports. This fact also hinders real time reporting capabilities. Similar to EHR, the development of RPP following an agile BI approach is limited in terms of diagnosis selection. The access to data source(s) determines what diagnoses could be monitored in reports. Moreover, as ETL is required, real time reporting is not supported. RPP1, developed using a custom web approach resulted in cumbersome data entry; ad-hoc diagnosis selection and ad-hoc data display. Therefore, the development team and technical expert reviewed this approach and determined a new development methodology and application architecture was needed that could address these application features (part of the adoption criteria). A small change in clinicians' monitoring needs made the RPP1 obsolete. Finally, application features in RPP that resulted from using a

structured CPMA methodology included optimized data entry, optimized diagnosis selection and optimized data display. The structured development methodology incorporates a systematic evaluation of usability as one of the key building blocks to define/adjust the performance model and configure the application at each iteration of development, which affects application features such as data entry, diagnosis selection and data display. Moreover, as the database schema used in the structured CPMA methodology is optimized for reporting, drill through reporting and near-real time were supported.

In Table 10 we evaluate how well each of the key evaluation criteria for the development of RPP is fulfilled by the four different approaches reviewed in this study.

Table 10: Related Works - Methodology

Methodology	**EHR**	**Agile BI**	**Custom Web Development**	**Structured CPMA Methodology**
Clinical performance (Goals, Metrics)	Ad-Hoc	Yes	Ad-Hoc	Yes
Clinical Acceptance (Adoption Criteria)	No	Moderate	No	Yes
Model Driven (Reporting Database and Clinical Dimensions)	No	Incomplete	No	Yes
Integrate Forms, Reports and Dimensions	No	Reports & Dimensions	Ad-hoc	Yes
Integrate Technical and Clinical Expertise	No	Yes	No	Yes
Systematic Evaluation of Usability	No	Use cases, user scenarios	Ad-hoc	Yes

Building RPP from an EHR is difficult. The main purpose of EHR systems is not performance monitoring. Any integrated report for performance monitoring is ad-hoc and not necessarily linked to metrics in a performance model. Moreover, EHR databases are transactional databases (relational schema), not optimized for reporting and not all the

necessary data to track clinical performance is available in EHR databases. Therefore, monitoring of goals is ad-hoc. ETL is required to generate monitoring reports; therefore, clinical dimensions are not mapped to forms and reports. Moreover, whether there is a systematic evaluation of usability during development is vendor dependent.

Agile BI falls short in data modeling. Not all clinical dimensions needed for performance monitoring can be mapped to the reporting database as the data model is restricted to the data available in external data sources (i.e. EHR, EMR, HIS), thus incomplete. Once ETL is done, clinical dimensions are integrated to monitoring reports in agile BI (monitoring reports are linked to goals and metrics). This approach improves upon traditional BI developments in that it integrates clinical and technical expertise during the development process, which leads to a better understanding of factors that affect adoption. However, the evaluation phase do not include interactions of end-users with the application, which limits insights on adoption criteria and rich information that can be obtained by the use of methods such as user experience walkthroughs, and the think alouds sessions. Agile BI uses user cases and user stories for usability testing.

Custom web developments are IT-centric. The focus is on application features rather than information needs; therefore, there is no emphasis on multi-disciplinary collaboration during the development of the application. The interaction between clinicians and the development team occurs at the beginning of the project and from that point on, developers are focused on the technical aspects of the application. The analysis of factors that impact clinical acceptance is very limited and changes are hard to implement (see Table 10). Changes in any of the application tiers will require changes in

the other tiers to ensure consistency. For example, in RPP one key adoption criterion was the time to complete a record in less than 30 seconds. In a custom Web development approach, this adoption criteria means that the code needs to be rewritten for the form, which generates a new ORM database and therefore reports need to be reconfigured to keep consistency in the application. Although possible, the incorporation of adoption criteria in this development approach is poor, and the integration of forms, reports and clinical dimensions is ad-hoc.

In the structured CPMA methodology, forms and reports are linked to the data model that maps to the goal model and metrics. Therefore, any change in the goal model that results in updates to the data model will automatically update forms and reports.

5.3. Practice Profile Applications

In this section, we compare the practice profile application, RPP that was developed according to our development methodology, with other practice profile applications built as custom web applications using the criteria identified in section 3.3.

RPP, JIT and LogMD are similar applications that collect data for monitoring clinical experience. RPP focuses on the experience of medical residents, JIT focuses on internal medicine students and, LogMD is for anesthesiologists. While all three applications were designed for simple, easy data entry and monitoring clinical performance, RPP is the one that meets most of the criteria in terms of engineering efforts and application features. Clinical administrators and developers can easily configure RPP. Values in dimensional data tables can be updated by clinicians and changes are

immediately reflected in forms dropdowns and reports, whereas in JIT and LogMD this is not possible. The RPP database schema is optimized for reporting and fully supports BI capabilities; drill through data and real time reporting are fully supported. JIT and LogMD do not support third-party reporting tools or drill through data in reports. Reports are static and although they can be exported to secondary applications, data needs to be processed (ETL) to fully support BI capabilities. JIT has a rigid longer data entry to ensure the same steps are followed for all students and ensure consistency in how students were assessed for a single clinical experience. RPP was designed for faster data entry. It was important for a resident to log all patient encounters in order to self-assess their entire clinical experience.

Table 11: Comparison of Practice Profile Applications

	JIT	**LogMD**	**RPP**
Engineering Effort			
Form Factors	All form factors	All form factors	All form factors
Form/Report Linkage	No	Yes	Yes
Third-party reporting tools	No. Hard Coded Reports	No. Hard Coded Reports	Yes. Dimensional Model
Application Configuration	None	None	Data tables. UI Controls
Repeatable	Limited	Limited	Yes
User Centered Design	No	Maybe	Yes
Application Features			
Data Entry	2-5 min.	< 2 min	<30 sec
Diagnosis Selection	One level.	Two levels.	Two levels.
Data Display (Templated Summaries)	No	No	Yes
Drill-Through Reporting	No	No	Yes
Near Real-Time reporting,	Yes	Yes	Yes
Security	Basic Login	Basic Login	Basic Login
Methodology			
Clinical performance (Goals, Metrics)	Yes	Yes	Yes
Clinical Acceptance (Adoption Criteria)	NA	NA	Yes

Model Driven (Reporting Database and Clinical Dimensions)	No	Yes	Yes
Integrate Forms, Reports and Dimensions	No	Yes	Yes
Integrate Technical and Clinical Expertise	NA	NA	Yes
Systematic Evaluation of Usability	After the fact	-	Yes

The development of RPP followed a structured development methodology that included systematic reviews of usability at each iteration of development. This helped identify key adoption criteria that could be addressed during the development of the technology. JIT evaluation of usability was done, via an anonymous survey (n=68 students), after the system was rolled out to 95 students (G. Ferenchick et al., 2008). The fact usability was not evaluated and addressed during the development process could explain why students asked for quicker access to information, search options, alphabetical listing of diagnoses, and a more comprehensive database after the system was deployed. All these factors are directly related to application features such as data entry, data display and diagnosis selection. Moreover, JIT evaluation of usability after deployment could explain why the reported levels of user satisfaction were less than 50% in ten out of the eleven categories evaluated (G. Ferenchick et al., 2008).

Chapter 6. Conclusions and Future work

6.1. Conclusions

In this thesis we presented an application development methodology for the development of CPMA. We confirmed the validity of our approach using a case study of a CPMA for performance monitoring of clinical training called Residence Practice Profile application. We believe our approach could be transferable to other practice profile applications and moreover, it should be useful for most form-based clinical monitoring and logging applications that collect data for performance monitoring. Our approach addresses the complexity of application development in healthcare environments, where data sources are often a stumbling block for monitoring performance, particularly when care is provided across multiple independent locations. Our development methodology is structured in such a way that constant changes in information needs (inherent to BI) can be addressed with ease by IT developers and clinical administrators.

The key factors that make our development methodology a better approach for the development of CPMA are:

1. A better definition of application requirements by defining metrics linked to goals, data sources and adoption criteria
2. The mapping of metrics to a multidimensional central database (star-schema) optimized for reporting and complex analysis of data, that eliminates the need of ETL processes, and enable real-time reporting and drill-through data

3. Configuration of application forms and reports linked through a reporting database that eliminates the ambiguity between forms and reports when configured separately
4. A systematic approach to evaluation using user experience walkthroughs at each iterations of development, that guide the technical and clinical experts in the creation of checklists to drive and prioritize the main innovations and close the gap that traditionally exist between clinicians and the IT development team.

Clinicians are actively co-creating their user experience whilst the development team and experts are getting a better understanding of what final users value most. And this is supported by an information technology framework that provides a combination of the right application configuration reflecting a better understanding of information technology to support clinical performance monitoring, and the definition of the application in terms of a performance model (definition of metrics linked to goals in terms of clinical dimensions mapped to reports and forms) which can be mapped directly to optimized application code that is easy to deploy and maintain. Most importantly the iterative methodology guides CPMA development to successful adoption by systematically walking through the user experience.

Our development methodology improves the application configuration and user experience for data collection and reporting when evaluated against EHR, custom web applications and agile BI approaches.

6.2. Limitations and Future Work

There are some limitations in this study. First, we used a single case study to confirm that our application development methodology is helpful to guide the development of CPMAs. More research is needed to confirm that the methodology is transferrable and can be used by other clinical residency programs to build their CPMA. Also, more research is needed to evaluate that the methodology can also be effective for developing any other CPMA, not only for the development of practice profiling applications. Second, we have not evaluated the impact that an application developed following our development methodology has in clinical practice. We have evaluated the appropriateness and effectiveness of a structured methodology that streamline the configuration of applications for performance monitoring and maximizes user's intent to adopt the technology, however, more research is needed to demonstrate the effectiveness of the resultant CPMA in monitoring clinical practice and its actual rate of use and adoption. Third, our application development methodology is focused on applications that balance simple, easy-to-use interfaces and powerful BI reporting capabilities, however, it requires double data entry since the CPMA is a new application that does not replace or use existing EMR/EHR applications. Future work is needed to developing an integrated performance monitoring system. Four, we assumed the storage needs in the CPMA is not compromised by restricting the database to a star-schema. Finally, we assumed that a CPMA is strictly a standalone application. More work is needed to address communication, service, and application integration issues.

References

Ammenwerth, E., Iller, C., & Mahler, C. (2006). IT-adoption and the interaction of task, technology and individuals: A fit framework and a case study. *BMC Medical Informatics and Decision Making, 6*(1), 3. doi:10.1186/1472-6947-6-3

Baarah, A., Kuziemsky, C., Chamney, A., Bindra, J., & Peyton, L. (2014). A design strategy for health information systems to address care process management. *Proceedings of the International Conference on Information and Communication Systems, IEEE Press,*

Baarah, A., Mouttham, A., & Peyton, L. (2012). Architecture of an event processing application for monitoring cardiac patient wait times. *International Journal of Information Technology and Web Engineering (IJITWE), 7*(1), 1-16. doi:10.4018/jitwe.2012010101

Banavar, G., Beck, J., Gluzberg, E., Munson, J., Sussman, J., & Zukowski, D. (2000). Challenges: An application model for pervasive computing. *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking,* 266-274.

Beck, K. (2000). *Extreme programming explained: Embrace change* Addison-Wesley Professional.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., . . . Jeffries, R. (2001). Manifesto for agile software development.

Benbunan-Fich, R. (2001). Using protocol analysis to evaluate the usability of a commercial web site. *Information & Management, 39*(2), 151-163.

Biju, S. M. (2008). Agile software development. *E-Learning and Digital Media, 5*(1), 97-102.

Blumenthal, D., & Tavenner, M. (2010). The "Meaningful use" regulation for electronic health records. *New England Journal of Medicine, 363*(6), 501-504.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology, 3*(2), 77-101.

Chaudhuri, S., & Dayal, U. (1997). An overview of data warehousing and OLAP technology. *SIGMOD Rec., 26*(1), 65-74.

Cohn, M. (2010). *Succeeding with agile: Software development using scrum* Pearson Education.

Colliat, G. (1996). OLAP, relational, and multidimensional database systems. *ACM Sigmod Record, 25*(3), 64-69.

Connolly, A., Davis, K., Casey, P., Keder, L., Pradhan, A., Page, R., ... & Dalrymple, J. (2010). Multicenter trial of the clinical activities tool to document the comparability of clinical experiences in Obstetrics–Gynecology clerkships. *Academic Medicine, 85*(4), 716-720.

Dan North & Associates.Introducing BDD. Retrieved from http://dannorth.net/introducing-bdd/

Davis, F., Bagozzi, R., & Warshaw, P. (1989). User acceptance of computer technology: A comparison of two theoretical models.*35*(8), 982-1003. doi:10.1287/mnsc.35.8.982

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319-340.

Denton, G. D., DeMott, C., Pangaro, L. N., & Hemmer, P. A. (2006). LITERATURE REVIEWS: Narrative review: Use of student-generated logbooks in undergraduate medical education. *Teaching and Learning in Medicine, 18*(2), 153-164.

Douglas, D. (2008). A critical examination of mixing methods for developing business and management theories. *7th European Conference on Research Methodology for Business and Management Studies: Ecrm 2008,* 137.

Dresner, H. (2008). *The performance management revolution: Business results through insight and action* John Wiley & Sons.

Duan, L., & Da Xu, L. (2012). Business intelligence for enterprise systems: A survey. *Industrial Informatics, IEEE Transactions On, 8*(3), 679-687.

Ellaway, R. H., Pusic, M. V., Galbraith, R. M., & Cameron, T. (2014). Developing the role of big data and analytics in health professional education. *Medical Teacher, 36*(3), 216-222.

Epstein, R. M., Siegel, D. J., & Silberman, J. (2008). Self-monitoring in clinical practice: A challenge for medical educators. *Journal of Continuing Education in the Health Professions, 28*(1), 5.

Eva, W., Kevin, & Regehr, W., Glenn. (2007). Knowing when to look it up: A new conception of self-assessment ability. *Academic Medicine. RIME: Proceedings of the Forty-Sixth Annual Conference November 4-November 7, 2007, , 82 (10)* S81-S84.

Ferenchick, G., Fetters, M., & Carse, A. M. (2008). Just in time: Technology to disseminate curriculum and manage educational requirements with mobile technology. *Teaching and Learning in Medicine, 20*(1), 44-52.

Ferenchick, G. S., Foreback, J., Towfiq, B., Kavanaugh, K., Solomon, D., & Mohmand, A. (2010). The implementation of a mobile problem-specific electronic CEX for assessing directly observed student-patient encounters. *Medical Education Online, 15*, 10.3402/meo.v15i0.4276. doi:10.3402/meo.v15i0.4276 [doi]

Ferenchick, G. S., & Solomon, D. (2013). Using cloud-based mobile technology for assessment of competencies among medical students. *PeerJ, 1*, e164. doi:10.7717/peerj.164

Ferenchick, G. S., Solomon, D., Foreback, J., Towfiq, B., Kavanaugh, K., Warbasse, L., . . . Mohmand, A. (2013). Mobile technology for the facilitation of direct observation and assessment of student performance. *Teaching and Learning in Medicine, 25*(4), 292-299. doi:10.1080/10401334.2013.827972

Fonteyn, M. E., Kuipers, B., & Grobe, S. J. (1993). A description of think aloud method and protocol analysis. *Qualitative Health Research, 3*(4), 430-441. doi:10.1177/104973239300300403

Forrester. (2015). Build an agile bi organization. Retrieved from http://blogs.forrester.com/boris_evelson/15-02-05-build_an_agile_bi_organization

Fowler, M. (2002). *Patterns of enterprise application architecture* Addison-Wesley Longman Publishing Co., Inc.

Gartner. (2013a). Gartner predict business intelligence and analytics will remain top focus for CIOs through 2017. Retrieved from http://www.gartner.com/newsroom/id/2637615

Gartner. (2013b). IT glossary - clinical data repositories. Retrieved from http://www.gartner.com/it-glossary/cdr-clinical-data-repository

Gartner. (2013c). IT glossary - EHR. Retrieved from http://www.gartner.com/it-glossary/ehr/

Gellersen, H., & Gaedke, M. (1999). Object-oriented web application development. *IEEE Internet Computing,* (1), 60-68.

Gunter, T. D., & Terry, N. P. (2005). The emergence of national electronic health record architectures in the united states and australia: Models, costs, and questions. *Journal of Medical Internet Research, 7*(1), e3. doi:10.2196/jmir.7.1.e3

Habib, J. L. (2010). EHRs, meaningful use, and a model EMR.(managed care matters). *Drug Benefit Trends, 22*(4), 99.

Heeks, R. (2005). Health information systems: Failure, success and improvisation. *International Journal of Medical Informatics, 75*(2), 125-137. doi:10.1016/j.ijmedinf.2005.07.024

Hibernate.HIbernate. everything data.

Holzinger, A. (2005). *Usability engineering methods for software developers* doi:10.1145/1039539.1039541

Iglar, K., Polsky, J., & Glazier, R. (2011). Using a web-based system to monitor practice profiles in primary care residency training. *Canadian Family Physician, 57*(9), 1030-1037.

Jaeger, R. G., & Halliday, T. R. (1998). On confirmatory versus exploratory research. *Herpetologica,* , S64-S66.

Johnson, R. (2005). J2EE development frameworks. *Computer, 38*(1), 107-110.

Kalakota, R., & Robinson, M. (1999). *E-business : Roadmap for success*. Reading, Mass.: Reading, Mass. : Addison-Wesley, c1999.

Kaplan, B., & Harris-Salamone, K. (2009). Health IT success and failure: Recommendations from literature and an AMIA workshop. *Journal of the American Medical Informatics Association : JAMIA, 16*(3), 291-299. doi:10.1197/jamia.M2997

Kendall, J. E., & Kendall, K. E. (2004). Agile methodologies and the lone systems analyst: When individual creativity and organizational goals collide in the global IT environment.(information technology). *Journal of Individual Employment Rights, 11*(4), 333-347.

Khan, K., & Ramachandran, S. (2012). Conceptual framework for performance assessment: Competency, competence and performance in the context of assessments in healthcare – deciphering the terminology. *Med Teach, 34*(11), 920-928. doi:10.3109/0142159X.2012.722707

Kho, A., Henderson, L., Dressler, D., & Kripalani, S. (2006). Use of handheld computers in medical education. *Journal of General Internal Medicine, 21*(5), 531-537. doi:10.1111/j.1525-1497.2006.00444.x

Kimball, R. (2008). *The data warehouse lifecycle toolkit* John Wiley & Sons.

Kimball, R. (2013). *The data warehouse toolkit the definitive guide to dimensional modeling, third edition* (3rd ed.. ed.). Indianapolis, Ind.: Indianapolis, Ind. : Wiley, c2013.

Kronz, A. (2006). Managing of process key performance indicators as part of the ARIS methodology. (pp. 31-44) Springer Berlin Heidelberg. doi:10.1007/3-540-30787-7_3

Larson, D. (2012). Agile methodologies for business intelligence. *Busness intelligence and agile methodologies for knowledge-based organizations: Cross-disciplinary applications* (pp. 101-119) ICI Global. doi:10.4018/978-1-61350-050-7

Lebas, M. J. (1995). Performance measurement and performance management. *International Journal of Production Economics, 41*(1), 23-35.

Leggat, S., Bartam, T., & Stanton, P. (2011). High performance work systems: The gap between policy and practice in health care. *Taken from Http://Www.Download.Bham.Ac.Uk/Hsmc/Sandraleggat.Pdf Last Accessed January, 31*

LogMD. Retrieved from http://www.logmd.com

Lyman, J., Schorling, J., Nadkarni, M., May, N., Scully, K., & Voss, J. (2008). Development of a web-based resident profiling tool to support training in practice-based learning and improvement. *Journal of General Internal Medicine, 23*(4), 485-488. doi:10.1007/s11606-007-0467-y

Martin, R. C. (2003). *Agile software development: Principles, patterns, and practices* Prentice Hall PTR.

Mata, P., Kuziemsky, C., Singh, J., Baarah, A., & Peyton, L. (2014). Engineering a performance management system to support community care delivery . *4th Symposium on Foundations of Health Information Engineering and Systems and 6th Workshop on Software Engineering in Healthcare,*

Mata, P., Baarah, A. H., Kuziemsky, C., & Peyton, L. (2014). An application meta-model for community care. *Procedia Computer Science; the 5th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2014)/ the 4th International Conference on Current and Future Trends of Information and Communication Technologies in Health(TRUNCATED), 37*(0), 465-472. doi:http://dx.doi.org/10.1016/j.procs.2014.08.070

Mata, P., Chamney, A., Viner, G., Archibald, D., & Peyton, L. (2015). A development framework for mobile healthcare monitoring apps.(report).*19*(3-4), 623.

Mettler, T., & Vimarlund, V. (2009). Understanding business intelligence in the context of healthcare. *Health Informatics Journal, 15*(3), 254-264. doi:10.1177/1460458209337446 [doi]

Microsoft. (2015). Overview of online analytical processing (OLAP). Retrieved from https://support.office.com/en-us/article/Overview-of-Online-Analytical-Processing-OLAP-15d2cdde-f70b-4277-b009-ed732b75fdd6#bmwhat_is_on_line_analytical_processing

Middleton, G., Peyton, L., Kuziemsky, C., & Eze, B. (2009). A framework for continuous compliance monitoring of ehealth processes. *Privacy, Security, Trust and the Management of E-Business, 2009. CONGRESS'09. World Congress On,* 152-160.

Minister of Public Works and Government Services Canada 2010. (2010). *Electronic health records in canada: An overview of federal and provincial uadit reports*. (). Ontario, Canada: Minister of Public Works and Government Services Canada 2010.

Novak, L., Brooks, J., Gadd, C., Anders, S., & Lorenzi, N. (2012). Mediating the intersections of organizational routines during the introduction of a health IT system. *European Journal of Information Systems, 21*(5), 552-569.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45-77.

Renshaw, S., Saywell, R. M., Burba, J., Butler, A. L., Zollinger, T., Kiovsky, R. D., . . . O'Hara, B. (2014). Trends in patient encounters: Implications for family medicine clerkships. *Family Medicine; Fam.Med., 46*(10), 761-769.

Rowley, P., Gough, R., Doylend, N., Thirkill, A., & Leicester, P. (2013). From smart homes to smart communities: Advanced data acquisition and analysis for improved sustainability and decision making. *Information Society (i-Society), 2013 International Conference On,* 263-268.

Sacks, M. (2012). In Debois P. (Ed.), *Pro website development and operations streamlining DevOps for large-scale websites*. New York]: New York : Apress, c2012.

Sclafani, J., Tirrell, T., & Franko, O. (2013). Mobile tablet use among academic physicians and trainees. *Journal of Medical Systems, 37*(1), 1-6. doi:10.1007/s10916-012-9903-6

Singh, I., & Johnson, M. (2002). *Designing enterprise applications with the J2EE platform* Addison-Wesley Professional.

Skolnik, N. S. (2011). In Neil S. Skolnik (Ed.), *Electronic medical records*. New York: Humana Press, Springer Science+Business Media. doi:10.1007/978-1-60761-606-1 Springer New York Dordrecht Heid

Soeken, M., Wille, R., & Drechsler, R. (2012). Assisted behavior driven development using natural language processing. *Objects, models, components, patterns* (pp. 269-287) Springer.

Tang Jun, Cui Kai, Feng Yu, & Tong Gang. (2009). The research & application of ETL tool in business intelligence project. *Information Technology and Applications, 2009. IFITA '09. International Forum On, , 2* 620-623. doi:10.1109/IFITA.2009.48

Tegegne, A., & Peyton, L. (2013). Application framework support for process-oriented software development. *International Journal of Electronic Business, 10*(3), 232-253.

Vredenburg, K., Mao, J., Smith, P. W., & Carey, T. (2002). A survey of user-centered design practice. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* 471-478.

Wallace, S., Clark, M., & White, J. (2012). 'It's on my iPhone': Attitudes to the use of mobile computing devices in medical education, a mixed-methods study. *BMJ Open, 2*(4) doi:10.1136/bmjopen-2012-001099

Warner, A., Bienfait, K., Bledsoe, M., Burckart, G., Flamion, B., Knoppers, B., . . . Uyama, Y. (2013). Improving clinical trial sampling for future research - an international approach: Outcomes and next steps from the DIA future use sampling workshop 2011. *Pharmacogenomics, 14*(1), 103-112. doi:10.2217/PGS.12.193

Waterson, P. (2014). Health information technology and sociotechnical systems: A progress report on recent developments within the UK national health service (NHS). *Applied Ergonomics, 45*(2), 150-161.

Watson, H. J. (2009). **Tutorial: Business intelligence – past, present, and future"**. *Communications of the Association for Information Systems, 25*(39)

WHO. (2015). WHO definition of palliative care. Retrieved from http://www.who.int/cancer/palliative/definition/en/

Wu, J., Wang, S., & Lin, L. (2007). Mobile computing acceptance factors in the healthcare industry: A structural equation model. *International Journal of Medical Informatics, 76*(1), 66-77.

Zeng, J., Jumbo, A., & Zhang, J. (2014). Embracing agile health analytics: A use case for stroke registry. *International Journal of Health Research and Innovation, 2*(2), 1-10.