Software Frameworks in Biomedical Informatics: A Systematic Review and Research Agenda

Mahmood Ghaleb Al-Bashayreh, Nor Laily Hashim, Ola Taiseer Khorma
University Utara Malaysia/School of Computing, Sintok, Kedah, Malaysia
Email: mahmood.g.al-bashayreh@ieee.org, laily@uum.edu.my, ola_khorma@hotmail.com

***Abstract*—Design of Mobile Patient Monitoring Frameworks (MPMF) is an emerging study in the biomedical informatics domain. However, literature about this topic remains fragmented: first, there are no categories in the literature to characterize these MPMF; second, there are no success factors that must be satisfied to enhance the design of these MPMF; third, there is a need for a research agenda that provides a foundation for further research and development to enhance the design of these MPMF. To address this gap, this paper reviews software frameworks in general and conducts a systematic review of 20 designed frameworks in the biomedical informatics domain. Consequently, this paper provides three results. First, it identifies a number of categories to be used to gain an intensive and extensive understanding of the designed MPMF. Second, it identifies a set of success factors, called Factors of Successful Application Frameworks (FSAF), to be used to enhance the design of MPMF. Third, it presents a research agenda that shows lacks and gaps in the designed MPMF and provides a foundation to help researchers design enhanced MPMF. The results show that, as of yet, there is no study integrating all the FSAF to enhance the design of MPMF. Therefore, there is a need for further research to enhance the existing MPMF design.**

***Index Terms*—Mobile Patient Monitoring Frameworks (MPMF), Mobile Patient Monitoring Systems (MPMS), Factors of Successful Application Frameworks (FSAF), Wireless sensors, framework design**

I. Introduction

Software reuse has been one of the most important foci of software engineering for decades [21]. Reuse is defined in the software engineering vocabulary as "building a software system at least partly from existing pieces to perform a new application"[23]. Among the primary software reuse techniques are [25]: architectural patterns [26], design patterns [27], component-based development [28], software product lines [29], program libraries [25], model-driven engineering [30], and software frameworks [25].

A software framework is an ideal reuse technique compared with others. It captures the essence of software engineering reuse techniques [31, 32] to achieve maximum [27] large-scale reuse [32, 33]. For instance, a framework allows the reuse of software design [21, 34] including both architectural and non-architectural designs [35]. A framework can reuse architectural design using patterns, while it can also reuse non-architectural design using components [21, 32]. Moreover, a software framework is one of the most suitable solutions to simplify application development and overcome their development complexities [31, 36]. In fact, software frameworks benefit application development and enhances overall software development quality [37]. For example, using software frameworks reduces development time [38], efforts [34], and cost [39]. Similarly, using software frameworks decreases line-of-code [38], increases developer productivity [40], and reduces maintenance efforts [41].

Consequently, Mobile Patient Monitoring Frameworks (MPMF) were introduced as a suitable solution to enhance the overall development quality and overcome the development complexity of Mobile Patient Monitoring Systems (MPMS) using wireless sensors [3, 19]. However, the literature related to these emerging MPMF remains fragmented. There is a shortage in designing MPMF. In addition, there is a need to enhance the design of these frameworks [3, 19]. First, there are no categories in the literature to characterize these MPMF. Second, there are no success factors that must be satisfied in enhancing the design of these MPMF. Finally, there is a need for a research agenda that provides a foundation for further research and development to enhance the design of these MPMF. Therefore, the objectives in this paper are as follows: first, to identify a number of categories to be used to gain an intensive and extensive understanding of the designed MPMF; second, to identify a set of success factors, called Factors of Successful Application Frameworks (FSAF), to be used to enhance the design of MPMF; third, to present a research agenda that shows lacks and gaps in the designed MPMF to provide a foundation to help researchers to design an enhanced MPMF.

The remainder of this paper is structured as follows: Section II introduces MPMS. The proposed research framework to satisfy the objectives of this paper is introduced in Section III. Section IV presents a systematic review of frameworks in the biomedical informatics domain. Finally, Section V concludes this paper and briefly discusses future work.

II. Mobile Patient Monitoring Systems

Patient monitoring systems are considered an applied research area of biomedical informatics [42] and among its earliest applications [43]. No doubt, these systems

doi:10.4304/jsw.8.12.2996-3006

have improved the quality of healthcare [19, 44, 45]. These systems do not replace the role of healthcare professionals; instead, they attempt to assist, complement their roles [19, 46], and provide an alternative of monitoring patients within the boundaries of healthcare organizations [47]. Therefore, these systems automate repeated or continuous tasks required for monitoring patients, focusing on adherence to medical advice and detection of abnormal health events, carrying out analysis, and informing healthcare professionals once abnormal health events are detected. Thus, assisting professionals to focus on providing experienced therapeutic intervention on time [46].

However, the emergence of mobile devices and wireless sensor technologies has inspired the researchers to study the potential of adopting these technologies to develop MPMS that satisfy the need for personal lifetime health monitoring systems [2, 48]. Mobile devices, such as personal digital assistants (PDA) and smartphones, obviously have contributed to the development of patient monitoring systems [48, 49]. For example, they provide a platform to develop MPMS [50, 51] and act as a base unit to collect biomedical data, such as vital signs, from wireless sensors [19, 49]. Similarly, wireless sensors, including wireless body sensor networks and wireless environmental sensors, have contributed significantly to the development of patient monitoring systems [3]. For instance, wireless body sensors can be used to monitor patient biomedical data (vital signs) such as blood pressure and body temperature, as well as monitor patient physical activities such as walking and running [19]. Another example, environmental sensors can be used to monitor the surrounding environmental conditions that affect the patients such as, air temperature, humidity, lighting level, and location [3].

III. Research Framework

To achieve the objectives of this paper, three research processes were conducted as shown in Fig. 1. These are; literature review, success factor identification, and lacks and gaps identification. The following subsections are a discussion of the steps and the methods used in these processes, in addition to their outcomes.

A. Literature Review Process

This process is intended to review the literature using two steps: literature search, then literature content analysis. The objective of the first step is to collect scholarly articles related to this research and document them in a bibliography database as the main outcome of this step. The objective of the second step is to characterize previous studies to comprehend the literature through identifying a set of categories as the main outcome of the literature review process. The following subsections are discussions of the methods used in these two steps.

Step 1) Searching Literature: In this step, the literature search method introduced by Brocke et al. [52], was used. This method was conducted in this research by customizing the literature review framework of this

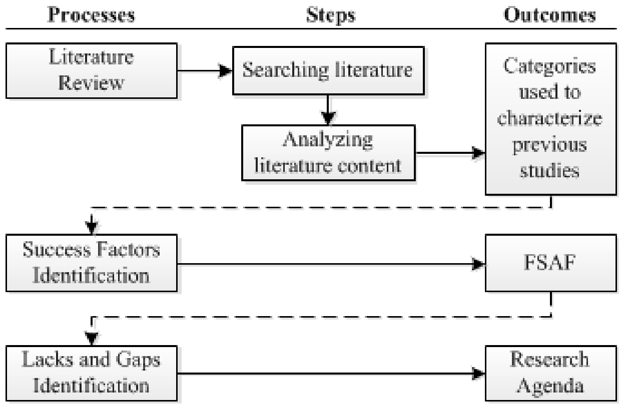

Figure 1. Research framework.

method. This framework includes three phases, which focus on searching and documenting the literature to provide a comprehensible and credible literature review process. This process is intended to increase researcher confidence in using current study outcomes in further research.

In the first phase, the literature scope was defined by identifying the following five characteristics. First, the focus of the literature search involved all scholarly articles related to design research. Second, the goal of the literature search was to identify the various waves of thoughts in academic research to design frameworks in biomedical informatics. Third, the perspective of the literature search was neutral, which means it does not reflect any certain opinion that supports a specific idea or principle. Fourth, the audience of the literature search results is specialized scholars designing frameworks in biomedical informatics. Fifth, the coverage of the literature search was a representative sample, which was selected based on specific criteria (i.e., year of publication and leading article source) to represent all research articles designing frameworks in biomedical informatics.

In the second phase, a set of key terms were identified, which are: design, application, framework, mobile, patient, monitoring, system, and sensors. In the third phase, the literature search process was conducted based on the identified key terms by focusing on scholarly articles from leading journals, conference proceedings, and scholarly databases. However, it was difficult to focus on a specialized range of journals because designing frameworks in the biomedical informatics domain is an interdisciplinary field of study that requires considering a wide range of articles. In fact, the range of journals dealing with designing such frameworks span biomedical informatics journals, mobile computing journals, information systems journals, communication journals, systems and software journals, software engineering journals, computer science journals, ubiquitous computing journals, and even network journals. Therefore, interdisciplinary online databases were chosen to begin searching.

Step 2) Analyzing Literature Content: The objective of content analysis is "to provide knowledge and understanding of the phenomenon under study" [53]. This research adopted an inductive analysis approach of content analysis proposed by Elo and Kyngäs [54]. This approach consists of three main phases: preparation, organizing and reporting.

In the preparation phase, two activities were performed. First, the unit of analysis is selected, which includes: requirements, application framework, framework design, framework architecture, patient monitoring systems, wireless sensors, and mobile technology. Second, the content data were read several times to make sense of the data in terms of the identified unit of analysis, to gain a comprehensive understanding and obtain a working knowledge of them.

In the organizing phase, five activities were conducted. First, open coding was performed by writing down notes to describe all aspects of the content. Second, the open coding data were collected and stored in a spreadsheet file. Third, the related data of the spreadsheet file initially were grouped based on observing the similarity among them. These groups were called subcategories. Fourth, the subcategories, which were identified in the previous activity, were organized under high-level categories. These high-level categories were called generic categories. Fifth, the generic categories, which were identified in the previous activity, were abstracted further based on the similarities or relations with other generic categories to provide a new abstract main category.

In the reporting phase, the analysis process was evaluated; then the reported analysis results were validated. First, the evaluation was performed by using tables to demonstrate a detailed explanation of all of the categories that were identified in the previous two phases. Second, the validation was performed by experts through publishing the categories in conference proceedings and discussing them with peers.

B. Success Factors Identification Process

The objective of this research process is to identify the FSAF as a main outcome by using a single step. In this step, a comparison and selection technique was proposed and used to compare the identified categories of factors that resulted from the previous process, to select the success factor among them. This proposed technique applies a pragmatic research approach [55]. Accordingly, a number of comparison criteria were identified and used as shown in Table I. These criteria give the first priority to satisfy framework extensibility and reusability since they have been proven as key characteristics to design successful frameworks [21, 38]. Aside from this, these criteria focus on any comparison criterion related to enhancing the overall design of MPMF to facilitate the development of MPMS.

Two alternative procedures were used in the comparison and selection technique. On one hand, whenever there are alternatives and only one must be selected, the following procedure is conducted. First, as shown in Table I, these alternatives are compared and weighted in terms of their support to a set of comparison criteria. Second, the alternatives that have zero weight are eliminated. Finally, the alternative that has the highest weight is selected. Alternatively, if there are different choices that can be considered together, the following procedure is conducted. All of the choices are selected.

C. Lacks and Gaps Identification Process

The objective of this research process is to identify the lacks and gaps in the literature using a single step. The objective of this step is to synthesize the previous studies, which designed frameworks in the biomedical informatics domain, using the identified FSAF. To meet this objective, this process uses the concept matrix technique that was introduced by Webster and Watson [56]. This matrix provides a method to organize, analyze, and synthesize previous studies, to develop a research agenda. This research agenda provides a foundation for the researchers to extend the state-of-the-art by bridging the gaps [52].

In this research, the concept matrix technique was customized as shown in the Table II. The first column in Table II lists the previous studies that represent the designed frameworks in the biomedical informatics domain. The following columns represent the identified FSAF. Each tick (✓) indicates that a specific study has satisfied a particular FSAF. The last column represents the total number of satisfied factors in each study out of the total number of all factors. The percentage row represents the percent of studies that have satisfied a particular factor, while the proportion row represents the number of these studied out of the total number of all studies.

IV. A SYSTEMATIC REVIEW

This section provides a systematic review of previous studies with frameworks designed in the biomedical informatics domain. First, it identifies a number of

TABLE I.
COMPARISON AND SELECTION TECHNIQUE

Comparison criteria	Categories		
	Subcategory(1)	…	Subcategory(n)
Criterion(1)	+1		0
Criterion (2)	0		+1
Criterion (3)	+1		+1
Criterion(m)	+1		0
Selection Result	3	…	2

TABLE II.
CONCEPT MATRIX

Previous studies	Success factors			Total
	Factor(1)	Factor(2)	Factor(f)	
Study(1)	✓	✓		2/f
Study(2)		✓		1/f
Study(3)				
Study(s)				
Percentages	25%	50%		
Proportions	1/s	2/s		

categories to be used to gain an intensive and extensive understanding of the designed MPMF. Second, it identifies a set of success factors called FSAF to be used to enhance the design of MPMF. Third, it presents a research agenda that shows the lacks and gaps in the designed MPMF, providing a foundation to help the researchers to design enhanced MPMF. This is achieved by following the three research processes, which were discussed earlier.

A. *Process 1: Literature Review*

By applying the two steps of the literature review process as elaborated earlier in Section III, many studies were collected and documented. Then, the contents of these studies were analyzed. At the end of this process, a set of categories was produced. These categories can be used to gain an intensive and extensive description of the current situation of the designed MPMF. For this purpose, 20 studies were analyzed and categorized based on the resulting categories that are shown in Table III. This research classified previous studies that designed frameworks in biomedical informatics into seven generic categories as shown in the second column in Table III.

TABLE III.
CATEGORIES USED TO CHARACTERIZE PREVIOUS STUDIES

Abstract main category	Generic categories	Subcategories	Related studies
Software frameworks in biomedical informatics domain	Scope	System infrastructure	[11]
		Middleware integration	[1, 5, 6, 10, 12-15]
		Enterprise application	[2-4, 7-9, 16-20]
	Application family	Patient monitoring systems	[2, 3, 7, 11, 13, 17, 19]
		Others	[1, 4-6, 8-10, 12, 14-16, 18, 20]
	Development approach	Object-based	[4, 9, 18]
		Component-based	[5, 8, 11, 12, 16, 19]
	Extensibility approach	White-box Framework (Inheritance)	[1, 3, 4, 10, 18]
		Black-box Framework (Composition)	[5, 8, 11, 12, 16, 19]
		Gray-box Framework (Mixed)	[9, 20]
	Evaluation	Extensibility	[4]
		Reusability	[7-9, 12, 18, 20]
		Other	[1-3, 6, 14, 16]
	Domain requirements	Support context awareness	[1, 3-6, 8-10, 13-16, 19, 20]
		Support anywhere anytime monitoring	[2, 6-8, 12-17, 19]
		Support real-time continuous monitoring	[1-3, 5, 8, 10, 12, 14, 17, 19]
		Support unlimited number of applications at run time	[1, 6-10, 14, 15, 19, 20, 22]
		Support unlimited number of sensors at design time	[1, 3-6, 8-10, 13-16, 19, 20]
	Platform	Mobile (smart phone or PDA)	[4, 8, 9, 11, 14, 16]
		Desktop (backend server)	[3, 10, 15]
		Other	[1, 2, 5, 7, 12, 13, 17-20]

B. *Process 2: Success Factors Identification*

This section presents the identified FSAF. To achieve this, the results from the previous process were used as an input to identify these factors. Identification of these factors is expected to enhance the design of MPMF. To identify these factors, a comparison and selection technique was proposed and used in this research as discussed in the success factors identification process in Section III. The results of applying the comparison and selection technique are illustrated in Table IV, which lists the selected FSAF. The comparison and selection technique for each factor is elaborated in the following subsections.

1) Scope - Enterprise Application Framework: Software frameworks are classified based on their scope, which are infrastructure, middleware integration, and enterprise application frameworks [21]. To identify a software framework scope that is suitable for this research, the comparison and selection technique was used as shown in Table V. The comparison criteria focus on designing an extensible and reusable framework. Aside from all of the benefits of these frameworks for systems development, the enterprise application frameworks (application frameworks for short) address software reuse in a radical way to support application business rules in a specific domain [37]. In essence, these application frameworks are strategic assets for software development companies. They directly focus on developing extensible end-user applications. Compared with infrastructure and middleware integration frameworks, application frameworks concentrate on facilitating and simplifying application development and overcoming their complexities [57]. Developing applications is complex in many domains including the biomedical informatics domain [10, 58, 59]. Therefore, developing application frameworks in biomedical informatics is an ideal solution to facilitate and simplify the development of these applications and overcome their complexities [3, 4, 7].

TABLE IV.
FACTORS OF SUCCESSFUL APPLICATION FRAMEWORKS

Abstract main category	Generic categories	Subcategories
Application frameworks to develop mobile patient monitoring systems using wireless sensors	Scope	Enterprise application frameworks
	Application family	Patient monitoring systems
	Development approach	Component-based
	Extensibility approach	Black box Framework (Composition)
	Evaluation	Extensibility
		Reusability
	Domain requirements	Support context awareness
		Support anywhere anytime monitoring
		Support real-time continuous monitoring
		Support unlimited number of applications at run time
		Support unlimited number of sensors at design time
	Platform	Mobile (smart phone or PDA)

2) Applications Family - Patient Monitoring Systems: Software frameworks can be designed specifically in biomedical informatics to develop a family of applications such as patient monitoring systems [60]. To identify a software framework application family that is suitable for this research, the comparison and selection technique was used as shown in Table VI. The comparison criteria focus on enhancing the design of patient-oriented monitoring systems in the biomedical informatics domain by considering the elders and patients with chronic disease as the primary stakeholders. Compared with other families of applications in biomedical informatics, patient monitoring systems have numerous benefits including enhancing healthcare quality [47] and reducing healthcare services costs [49]. For example, they provide elders and patients who suffer from chronic diseases with long-term monitoring [46], while being at their homes and during their everyday events [61]. Moreover, the flexibility and mobility provided by these systems (e.g. being monitored anywhere and anytime) have enhanced patient lifestyles in terms of increasing their dependability [45]. Aside from this, they encourage patients to take care of themselves and comply with their treatments [19, 46]. Therefore, designing application frameworks to support and enhance the development of patient monitoring systems is required to achieve all of the above-mentioned benefits [3, 7, 13].

TABLE V.
SCOPE COMPARISON AND SELECTION

Comparison criteria	Scope		
	Enterprise application	Middleware integration	System infrastructure
Support application business rules	+1	0	0
Support domain-specific applications	+1	0	0
Support software reuse	+1	0	0
Support extending end-user applications	+1	0	0
Simplify end-user applications development	+1	0	0
Selection result	5	0	0

TABLE VI.
APPLICATION FAMILY COMPARISON AND SELECTION

Comparison Criteria	Application families	
	Patient monitoring systems	Others
Support long-term monitoring	+1	0
Support anywhere/anytime monitoring	+1	0
Increasing patients dependability	+1	+1
Enhance patients treatments	+1	+1
Reduce healthcare services cost	+1	+1
Selection result	5	3

3) Development Approach - Component-based: Software frameworks can also be classified based on their development approaches [62]. The most common development approaches are object-oriented and component-based development. To identify a framework development approach that is suitable for this research, the comparison and selection technique was used as shown in Table VII. The comparison criteria focus on enhancing the design of an extensible and reusable framework. Compared with an object-oriented approach, component-based development provides more a formal and systematic approach to achieve framework modularity, which improves framework reusability [63]. For example, components support a large unit of reuse that can be plugged into a framework [39]. In addition, a component-based approach enhances the frameworks extensibility. For instance, it can support dynamic composition of component instances at run-time [5]. Furthermore, component-based frameworks support easy composition of third-party components [39]. However,

TABLE VII.
DEVELOPMENT APPROACH COMPARISON AND SELECTION

Comparison criteria	Development approaches	
	Object-oriented	Component-based
Support framework modularity	+1	+1
Support framework reusability	+1	+1
Support large unit of reuse	0	+1
Enhance framework extensibility	+1	+1
Support dynamic extensibility	0	+1
Support easy extensibility	0	+1
Selection result	3	6

the most important benefits of application frameworks originate from their reusability and extensibility [21], which are identified in [38] as the key characteristics of successful frameworks. Therefore, developing application frameworks using a component-based approach is the ideal technique to satisfy these key characteristics of successful frameworks [8, 11, 12].

4) Extensibility Approach - Black-box Framework: Frameworks can be classified based on their extensibility techniques into white-box, black-box, and gray-box frameworks. The Gray-box frameworks combine the first two techniques [21]. To identify a software framework extensibility approach that is suitable for this research, the comparison and selection technique was used as shown in Table VIII. The comparison criteria focus on enhancing the design of an extensible and reusable framework. Compared with white-box frameworks, black-box frameworks are reused and extended through plugging in components. The process of plugging in components is called composition [64]. Composition is the primary instantiation technique of black-box frameworks [34]. In fact, the composition technique is rooted in a component-based development approach [28, 39]. Application developers use this technique to reuse and extend black-box frameworks with no programming [34, 64]. In addition, they do not have to learn or look into the implementation of these frameworks [21, 39]. Hence, the application developers, including the beginners, found black-box frameworks easier to learn and use. Accordingly, black-box frameworks improve framework reusability. Furthermore, application developers can perform dynamic composition at run-time [28]. Therefore, black-box frameworks also improve framework extensibility [39]. Accordingly, framework experts prefer using black-box frameworks [65]. However, the application frameworks are commonly black-box frameworks [39]. Therefore, developing black-box frameworks is the ideal approach to satisfy the key characteristics of successful frameworks, which are reusable and extensible as well as to develop a family of applications in a specific domain [5, 8, 11].

5) Evaluation - Extensibility and Reusability: According to Fayad et al. [21], there is no complete framework that has all of the required features. Thus, framework extensibility is more important to insure the framework capability to be evolved and extended with new features. In spite of the advancement in software, application design, implementation, and maintenance are complex [57]. However, frameworks are among the most suitable solutions to simplify application development and overcome complexity [36]. This is because frameworks represent the fundamental reuse techniques of software engineering [31]. Framework reusability supports the domain knowledge and previous development of experts to avoid rebuilding applications from scratch. Therefore, it is required to ensure framework reusability for developing new applications [21]. To identify a framework evaluation approach that is suitable for this research, the comparison and selection technique was used as shown in Table IX. The comparison criteria focus on enhancing the design of an extensible and reusable framework. Accordingly, it is important to evaluate the key characteristics of successful frameworks such as extensibility [4] and reusability [7, 8], rather than evaluating framework functionalities [21].

6) Domain Requirements: There are five domain requirements, which are identified as subcategories in Table III, used to characterize designed frameworks in previous studies. These domain requirements can be considered together. Hence, the second procedure of the comparison and selection technique was followed by selecting all of the identified domain requirements. The justification for selecting each of these domain requirements is debated in the following subsections.

a) Context-Aware Monitoring: Context awareness in MPMS using wireless sensors allows effective detection of patient medical situations (e.g. high BP) based on patient contextual information (e.g. BP and dizziness). Accordingly, these systems can change behavior by adapting to the changes of a patient's medical situation, for example, through triggering an alarm [3, 66]. Therefore, adopting context awareness to develop MPMS using wireless sensors is required for monitoring patients in the biomedical informatics domain [13, 19].

b) Anywhere and Anytime Monitoring: Monitoring patients anywhere and anytime allows detecting their abnormal health events instantly, which allows patient monitoring systems to react immediately. For example, these systems can call healthcare professionals to enable

TABLE VIII.
EXTENSIBILITY APPROACH COMPARISON AND SELECTION

Comparison criteria	Extensibility approaches		
	White-box	Black-box	Gray-box
Support component-based framework	0	+1	+1
Enhance framework extensibility	+1	+1	+1
Enhance framework reusability	+1	+1	+1
Support dynamic extensibility	0	+1	+1
Support easy extensibility	0	+1	0
Easy to learn and use	0	+1	0
Extend with no programming	0	+1	0
Selection result	2	7	4

TABLE IX.
EVALUATION APPROACH COMPARISON AND SELECTION

Comparison criteria	Evaluation approaches		
	Extensibility	Reusability	Others
Ensure framework extensibility	+1	+1	0
Ensure framework reusability	+1	+1	0
Ensure framework evolution	+1	+1	0
Ensure ease of development	+1	+1	0
Selection result	4	4	0

them to make suitable clinical decisions [19, 46]. In addition, monitoring patients anywhere and anytime can improve their lifestyles by becoming more independent, more flexible and mobile while being monitored [45-47]. Therefore, anywhere and anytime monitoring is required for monitoring patients in biomedical informatics [2, 6, 12].

c) Real-Time Continuous Monitoring: Real-time continuous patient monitoring allows instant detection of patients' abnormal health events [19, 46]. Similar to anywhere and anytime patient monitoring, real-time continuous monitoring allows patient monitoring systems to react immediately. For example, the system can call healthcare professionals to enable them to make suitable clinical decisions [19, 46]. Accordingly, this provides proactive medical care to protect patients from future complications [19, 67] especially those who suffer from chronic diseases [68]. Therefore, real-time continuous patient monitoring is required for monitoring patients in biomedical informatics [1, 8, 10].

d) Support Unlimited Monitoring Applications at Design-Time: Elders, especially those who suffer from chronic diseases, need to be monitored by different dedicated applications such as monitoring hypertension and diabetes [68]. Therefore, supporting an unlimited number of applications to be developed at design time is required for monitoring patients in biomedical informatics [1, 6, 10].

e) Adding Unlimited Number of Monitoring Sensors at Design-Time: Sensors play a primary role in supporting patient monitoring systems [67]. In fact, the more the sensors are used; the more the comprehensive information can be gained through combining the data from these sensors. This enhances the detection efficiency of a patient's medical situation [10]. Therefore, supporting an unlimited number of sensors at design time is required for monitoring patients in biomedical informatics [1, 8, 9].

7) Mobile Platform: Frameworks can be designed for a specific platform, such as desktop (e.g. backend server) [3] and mobile platforms (e.g. smartphone) [8]. To identify a framework platform that is suitable for this research, the comparison and selection technique was used as shown in Table X. The comparison criteria focus on enhancing the design of efficient and accurate MPMS

TABLE X.
PLATFORM COMPARISON AND SELECTION

Comparison criteria	Framework platforms		
	Desktop	Mobile	Distributed
Support continuous monitoring	0	+1	+1
Support real-time monitoring	0	+1	0
Support long-term monitoring	+1	+1	0
Support anywhere/anytime monitoring	0	+1	0
Support cost effective monitoring	0	+1	0
Support context-aware monitoring	0	+1	+1
Selection result	1	6	2

in biomedical informatics, by considering the need for continuous, real-time, long-term, anywhere and anytime patient monitoring. Undoubtedly, the mobile platform supports portability and mobility in general [5, 19]. Compared with the desktop platform (backend server), the mobile platform (smart phone) has obvious benefits to be used for running an application framework to develop MPMS. In fact, the technological advancement of mobile devices in terms of hardware and software provides the required computations to monitor a patient without being connected to a backend server. Among these advancements are: processing and wireless capabilities, operating systems, multithreading ability, and storage capacity [7, 69]. Aside from this, it provides the required computations to extract patients' contextual information from context sources with sufficient accuracy [13]. The mobile platform also supports real-time patient monitoring [4, 5]. In this case, it can support context awareness and adaptation through direct detection of context changes [8]. Nevertheless, it supports privacy protection of the patient contextual data [8]. Moreover, the mobile platform can support patient monitoring anywhere and anytime [70]. In addition, it supports active (always turned on) continuous monitoring [7]. Subsequently, this provides proactive monitoring; early detection of abnormal health situations [71]. As well, it enables patients to monitor themselves during their daily life activities without interruption [70]. Furthermore, the mobile platform is adequate to avoid the continuous network communication costs required to transmit the data to a backend server [13]. Therefore, developing application frameworks in biomedical informatics hosted on a mobile platform is the ideal solution to achieve the mentioned benefits [4, 8, 9].

C. Process 3: Lacks and Gaps Identification

This section presents the process used to identify the lacks and gaps in previous studies that designed software frameworks in biomedical informatics. This was achieved by applying the lacks and gaps identification process that was elaborated in Section III. The main input for this process was the identified FSAF to design MPMF. These factors were used in this process to analyze 20 studies to identify the lacks and gaps in terms of percentages and proportions as shown in Table XI. The percentages and proportions for each factor were interpreted for analysis as a research trend (from 50% to 100%) or (10 to 20 out of 20), lack (from 1% to 49%) or (1 to 9 out of 20), and gap (0%) or (0 out of 20).

With reference to Table XI, it can be seen that there is a trend among the previous studies on addressing six of the FSAF. First, 55% or 11 out of 20 of the studies designed enterprise application frameworks to support the business rules of the applications in biomedical informatics. Second, 70% or 14 out of 20 of the studies adopted context awareness computing in their framework design. Third, 55% or 11 out of 20 of the studies designed their frameworks to support anywhere and anytime patient monitoring. Fourth, 50% or 10 out of 20 of the studies designed their frameworks to support real-time continuous monitoring. Fifth, 55% or 11 of 20 of the

TABLE XI.
PERCENTAGES AND PROPORTIONS OF PREVIOUS STUDIES THAT SATISFY THE IDENTIFIED FSCAF

Previous studies	Scope	Application family	Development approach	Extensibility approach	Evaluation		Domain requirements					Platform	Total
	Enterprise application	Patient monitoring systems	Component-based framework	Black-box framework	Extensibility	Reusability	Context awareness	Anywhere and anytime monitoring	Real-time continuous monitoring	Unlimited number of applications at run-time	Unlimited number of sensors at design time	Mobile (e.g. Smart phone)	
[1]							✓		✓	✓	✓		4/12
[2]	✓	✓						✓	✓				4/12
[3]	✓	✓					✓		✓		✓		5/12
[4]	✓				✓		✓				✓	✓	5/12
[5]			✓	✓			✓		✓		✓		5/12
[6]							✓	✓		✓	✓		4/12
[7]	✓	✓				✓		✓		✓	✓		6/12
[8]	✓		✓	✓		✓	✓	✓	✓	✓	✓	✓	10/12
[9]	✓					✓	✓			✓	✓	✓	6/12
[10]							✓		✓	✓	✓		4/12
[11]		✓	✓	✓								✓	4/12
[12]			✓	✓		✓		✓	✓		✓		6/12
[16]	✓		✓	✓			✓	✓				✓	6/12
[17]	✓	✓						✓	✓	✓	✓		6/12
[24]		✓					✓	✓			✓		4/12
[18]	✓					✓					✓		3/12
[14]							✓	✓	✓	✓	✓	✓	6/12
[19]	✓	✓	✓	✓			✓	✓	✓	✓	✓		9/12
[15]							✓	✓		✓	✓		4/12
[20]	✓					✓	✓			✓	✓		5/12
Percentages	55%	35%	30%	30%	5%	30%	70%	55%	50%	55%	85%	30%	
Proportions	11/20	7/20	6/20	6/20	1/20	6/20	14/20	11/20	10/20	11/20	17/20	6/20	

studies designed their frameworks to support an unlimited number of monitoring applications at design time. Last, 85% or 17 of 20 of the studies designed their frameworks to support an unlimited number of sensors.

Moreover, it clearly can be seen that there are lacks among the previous studies on addressing six of the FSAF. First, at most, 35% or 7 of 20 of the studies designed their frameworks to be dedicated to a patient monitoring application family. Second, exactly 30% or 6 of 20 of the studies designed their frameworks using a component-based approach. Third, just 30% or 6 out of 20 of the studies designed their frameworks to be instantiated as a black-box framework. Fourth, only 5% or 1 of 20 of the studies designed their frameworks to be evaluated in terms of extensibility. Fifth, just 30% or 6 of 20 of the studies designed their frameworks to be evaluated in terms of reusability. Last, exactly 30% or 6 of 20 of the studies designed their frameworks to be hosted and executed completely on a mobile platform.

In summary, it was found that there is a consensus among previous studies considering the following six factors from the FSAF: designed enterprise application frameworks as framework scope, consider context awareness computing, support anywhere and anytime patient monitoring, support real-time continuous monitoring, support unlimited number of monitoring applications, and support unlimited number of sensors as domain requirements. In addition, it was found that there are common lacks in considering the following six factors from the FSAF: design patient monitoring frameworks, use component-based development approach, use black-box framework extensibility approach, use extensibility and reusability evaluation approaches, and design frameworks to be hosted and completely executed on a mobile platform. However, based on the total number of satisfied factors in each study, it was found that the highest totals are 9 and 10 out of 12, which were only achieved by the studies of Broens et al. [19] and Kang et al. [8] respectively.

V. Conclusion and Future Work

This paper is a discussion of ongoing research on designing a MPMF. It begins by introducing the concept of software reuse. Then, it highlights the need for designing MPMF as an ideal reuse technique to develop MPMS. It also provides the research framework that was used to satisfy the research objectives of this paper. A systematic review of previous studies that designed frameworks in the biomedical informatics domain is presented, including the process of literature review, success factors identification, and lacks and gaps identification. The results show that there are few studies that designed MPMF. Aside from this, they have a severe lack in considering the identified FSAF. Moreover, there is no study that integrates all of the FSAF to design MPMF. Therefore, there is a need to bridge the gap between application framework designs and MPMS through designing of MPMF that satisfies all of the identified FSAF. In the future, the researchers attempt to integrate all of the FSAF in an enhanced design of MPMF, to help software engineers and developers to design and develop various MPMS easily for different diseases and with minimal cost.

References

[1] M. D. Rodríguez and J. Favela, "Assessing the Salsa Architecture for Developing Agent-Based Ambient Computing Applications," *Science of Computer Programming,* vol. 77, pp. 46-65, January 2012. doi: 10.1016/j.scico.2010.12.003

[2] V. Villarreal, G. Urzaiz, R. Hervas, and J. Bravo, "Monitoring Architecture to Collect Measurement Data and Medical Patient Control through Mobile Devices," in *Proc. 5th Int. Symp. Ubiquitous Computing and Ambient Intelligence*, Riviera Maya, Mexico, 2011.

[3] F. Paganelli and D. Giuli, "An Ontology-Based System for Context-Aware and Configurable Services to Support Home-Based Continuous Care," *IEEE Trans. Inf. Technol. Biomed.,* vol. 15, pp. 324-333, March 2011. doi: 10.1109/TITB.2010.2091649

[4] M. J. Mitchell, C. Meyers, A. A. Wang, and G. Tyson, "Contextprovider: Context Awareness for Medical Monitoring Applications," in *Proc. 33rd Annu. Int. Conf. IEEE Engineering in Medicine and Biology Society*, Boston, MA, 2011, pp. 5244-5247. doi: 10.1109/IEMBS.2011.6091297

[5] J. Lewandowski, H. E. Arochena, R. N. G. Naguib, and K. Chao, "A Portable Framework Design to Support User Context Aware Augmented Reality Applications," in *Proc. 3rd Int. Conf. Games and Virtual Worlds for Serious Applications*, Athens, Greece, 2011, pp. 144-147. doi: 10.1109/VS-GAMES.2011.28

[6] N. Roy, T. Gu, and S. K. Das, "Supporting Pervasive Computing Applications with Active Context Fusion and Semantic Context Delivery," *Pervasive and Mobile Computing,* vol. 6, pp. 21-42, February 2010. doi: 10.1016/j.pmcj.2009.10.002

[7] V. G. Koutkias, I. Chouvarda, A. Triantafyllidis, A. Malousi, G. D. Giaglis, and N. Maglaveras, "A Personalized Framework for Medication Treatment Management in Chronic Care," *IEEE Trans. Inf. Technol. Biomed.,* vol. 14, pp. 464-472, March 2010. doi: 10.1109/TITB.2009.2036367

[8] S. Kang, J. Lee, H. Jang, Y. Lee, S. Park, and J. Song, "A Scalable and Energy-Efficient Context Monitoring Framework for Mobile Personal Sensor Networks," *IEEE Trans. Mobile Computing,* vol. 9, pp. 686-702, August 2010. doi: 10.1109/TMC.2009.154

[9] A. Fortier, G. Rossi, S. E. Gordillo, and C. Challiol, "Dealing with Variability in Context-Aware Mobile Software," *Journal of Systems and Software,* vol. 83, pp. 915-936, November 2010. doi: 10.1016/j.jss.2009.11.002

[10] A. Esposito, L. Tarricone, M. Zappatore, L. Catarinucci, and R. Colella, "A Framework for Context-Aware Home-Health Monitoring," *International Journal of Autonomous and Adaptive Communications Systems,* vol. 3, pp. 75-91, December 2010. doi: 10.1504/IJAACS.2010.030313

[11] F. Balagtas-Fernandez, M. Tafelmayer, and H. Hussmann, "Mobia Modeler: Easing the Creation Process of Mobile Applications for Non-Technical Users," in *Proc. 15th Int. Conf. Intelligent User Interfaces*, Hong Kong, China, 2010, pp. 269-272. doi: 10.1145/1719970.1720008

[12] A. B. Waluyo, I. Pek, X. Chen, and W. Yeoh, "Design and Evaluation of Lightweight Middleware for Personal Wireless Body Area Network," *Personal and Ubiquitous Computing,* vol. 13, pp. 509-525, April 2009. doi: 10.1007/s00779-009-0222-y

[13] I. Mohomed, A. Misra, M. Ebling, and W. Jerome, "Context-Aware and Personalized Event Filtering for Low-Overhead Continuous Remote Health Monitoring," in *Proc. 9th IEEE Int. Symp. World of Wireless, Mobile and Multimedia Networks*, Newport Beach, CA, 2008, pp. 1-8. doi: 10.1109/WOWMOM.2008.4594820

[14] E. J. Ko, H. J. Lee, and J. W. Lee, "Ontology-Based Context Modeling and Reasoning for U-Healthcare," *IEICE Transactions on Information and Systems,* vol. E90-D, pp. 1262-1270, August 2007. doi: 10.1093/ietisy/e90-d.8.1262

[15] D. Zhang, Z. Yu, and C. Chin, "Context-Aware Infrastructure for Personalized Healthcare," in *Personalised Health Management Systems: The Integration of Innovative Sensing, Textile, Information and Communication Technologies*. vol. 117, C. D. Nugent, P. J. McCullagh, E. T. McAdams, and A. Lymberis, Eds., ed Washington, DC: IOS Press, 2005, pp. 154-163.

[16] F. C. Delicato, I. L. A. Santos, P. F. Pires, A. L. S. Oliveira, T. Batista, and L. Pìrmez, "Using Aspects and Dynamic Composition to Provide Context-Aware Adaptation for Mobile Applications," in *Proc. ACM Symp. Applied Computing*, Honolulu, HI, 2009, pp. 456-460. doi: 10.1145/1529282.1529381

[17] N. F. Ahmad, D. B. Hoang, and M. H. Phung, "Robust Preprocessing for Health Care Monitoring Framework," in *Proc. 11th IEEE Int. Conf. e-Health Networking, Applications and Services*, Sydney, Australia, 2009, pp. 169-174. doi: 10.1109/HEALTH.2009.5406196

[18] T. Laakko, J. Leppänen, J. Lähteenmäki, and A. Nummiaho, "Mobile Health and Wellness Application Framework," *Methods of Information in Medicine,* vol. 47, pp. 217-222, May/June 2008. doi: 10.3414/ME9113

[19] T. Broens, A. V. Halteren, M. V. Sinderen, and K. Wac, "Towards an Application Framework for Context-Aware M-Health Applications," *International Journal of Internet Protocol Technology,* vol. 2, pp. 109-116, 2007. doi: 10.1504/IJIPT.2007.012374

[20] J. E. Bardram and T. R. Hansen, "The Aware Architecture: Supporting Context-Mediated Social Awareness in Mobile Cooperation," in *Proc. 16th ACM Conf. Computer*

Supported Cooperative Work, Chicago, IL, 2004, pp. 192-201. doi: 10.1145/1031607.1031639

[21] M. Fayad, D. C. Schmidt, and R. E. Johnson, "Application Frameworks," in *Building Application Frameworks: Object-Oriented Foundations of Framework Design*, M. Fayad, D. C. Schmidt, and R. E. Johnson, Eds., ed New York, NY: Wiley, 1999, pp. 3-28.

[22] N. F. Ahmad and D. B. Hoang, "Assistive Health Care Monitoring Framework Using Active Database Approach," in *Proc. IADIS Int. Conf. e-Health*, Algarve, Portugal, 2009, pp. 19-26.

[23] ISO, IEC, and IEEE, *Systems and Software Engineering -- Vocabulary*. ISO/IEC/IEEE 24765:2010(E)) Piscataway, NJ: IEEE computer society, 2010.

[24] I. Mohomed, A. Misra, M. Ebling, and W. Jerome, "Harmoni: Context-Aware Filtering of Sensor Data for Continuous Remote Health Monitoring," in *Proc. 6th Annu. IEEE Int. Conf. Pervasive Computing and Communications*, Hong Kong, China, 2008, pp. 248-251. doi: 10.1109/PERCOM.2008.110

[25] I. Sommerville, *Software Engineering*, 9th ed. Boston, MA: Pearson, 2011.

[26] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal, *Pattern-Oriented Software Architecture : A System of Patterns* vol. 1st. Chichester, England: Wiley, 1996.

[27] E. Gamma, R. Helm, R. Johnson, and J. M. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*: Addison-Wesley, 1995.

[28] C. Szyperski, *Component Software: Beyond Object-Oriented Programming*, 2nd ed. New York, NY: Addison-Wesley, 2011.

[29] K. Pohl, G. Böckle, and F. v. d. Linden, *Software Product Line Engineering: Foundations, Principles, and Techniques*. Berlin, Germany: Springer, 2005.

[30] D. Gašević, D. Djurić, and V. Devedžić, *Model Driven Engineering and Ontology Development*, 2ed ed. New York, NY: Springer, 2009.

[31] M. Fayad, D. S. Hamu, and D. Brugali, "Enterprise Frameworks Characteristics, Criteria, and Challenges," *Communications of the ACM,* vol. 43, pp. 39-46, October 2000. doi: 10.1145/352183.352200

[32] W. Zhang and M. Kim, "What Works and What Does Not: An Analysis of Application Frameworks Technology," *Journal of Business Systems, Governance and Ethics,* vol. 1, pp. 15-26, November 2006.

[33] D. C. Schmidt and F. Buschmann, "Patterns, Frameworks, and Middleware: Their Synergistic Relationships," in *Proc. 25th Int. Conf. Software Engineering*, Portland, OR, 2003, pp. 694-704.

[34] M. Mattsson, "Comparison of Three Evaluation Methods for Object-Oriented Framework Evolution," in *Software Evolution and Feedback: Theory and Practice*, N. H. Madhavji, J. C. Fernández-Ramil, and D. E. Perry, Eds., ed West Sussex, England: Wiley, 2006, pp. 281-312.

[35] P. Clements, F. Bachmann, L. Bass, D. Garlan, J. Ivers, R. Little*, et al.*, *Documenting Software Architectures: Views and Beyond*, 2 ed.: Addison Wesley, 2011.

[36] J. v. Gurp and J. Bosch, "Role-Based Component Engineering," in *Building Reliable Component-Based Software Systems*, I. Crnkovic and M. Larsson, Eds., ed Norwood, MA: Artech House, 2002, pp. 135-154.

[37] J. Al-Dallal and P. Sorenson, "Reusing Class-Based Test Cases for Testing Object-Oriented Framework Interface Classes," *Journal of Software Maintenance and Evolution: Research and Practice,* vol. 17, pp. 169-196, May/June 2005. doi: 10.1002/smr.308

[38] R. Neumann, S. Günther, and N. Zenker, "Reengineering Deprecated Component Frameworks: A Case Study of the Microsoft Foundation Classes " in *Proc. 9th Int. Conf. Business Informatics*, Vienna, Austria, 2009, pp. 737-748.

[39] D. Parsons, A. Rashid, A. Telea, and A. Speck, "An Architectural Pattern for Designing Component-Based Application Frameworks," *Software—Practice and Experience,* vol. 36, pp. 157-190, November 2006. doi: 10.1002/spe.v36:2

[40] U. Kulesza, V. Alves, A. Garcia, C. J. P. d. Lucena, and P. Borba, "Improving Extensibility of Object-Oriented Frameworks with Aspect-Oriented Programming," in *Reuse of Off-the-Shelf Components*. vol. 4039, M. Morisio, Ed., ed Berlin, Germany: Springer, 2006, pp. 231-245.

[41] D. C. Schmidt, A. Gokhale, and B. Natarajan, "Leveraging Application Frameworks," *Queue,* vol. 2, pp. 66-75, July/August 2004. doi: 10.1145/1016998.1017005

[42] E. H. Shortliffe and M. S. Blois, "The Computer Meets Medicine and Biology: Emergence of a Discipline," in *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*, E. H. Shortliffe and J. J. Cimino, Eds., 3rd ed New York, NY: Springer, 2006, pp. 3-45.

[43] G. Wiederhold and E. H. Shortliffe, "System Design and Engineering in Health Care," in *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*, E. H. Shortliffe and J. J. Cimino, Eds., 3rd ed New York, NY: Springer, 2006, pp. 233-262.

[44] J. Sriram, M. Shin, D. Kotz, A. Rajan, M. Sastry, and M. Yarvis, "Challenges in Data Quality Assurance in Pervasive Health Monitoring Systems," in *Future of Trust in Computing*, D. Gawrock, H. Reimer, A. Sadeghi, and C. Vishik, Eds., ed Berlin, Germany: Vieweg+Teubner, 2009, pp. 129-142.

[45] I. Martínez, J. Escayola, M. Martínez-Espronceda, L. Serrano, J. D. Trigo, S. Led*, et al.*, "Standard-Based Middleware Platform for Medical Sensor Networks and U-Health," in *Proc. 17th Int. Conf. Computer Communications and Networks*, St. Thomas, US Virgin Islands, 2008, pp. 1-6. doi: 10.1109/ICCCN.2008.ECP.135

[46] S. Sneha and U. Varshney, "Enabling Ubiquitous Patient Monitoring: Model, Decision Protocols, Opportunities and Challenges," *Decision Support Systems,* vol. 46, pp. 606-619, November 2009. doi: 10.1016/j.dss.2008.11.014

[47] T. Bratan and M. Clarke, "Optimum Design of Remote Patient Monitoring Systems," in *Proc. 28th IEEE EMBS Annu. Int. Conf.*, New York, NY, 2006, pp. 6465-6468. doi: 10.1109/IEMBS.2006.260729

[48] Y. Ren, R. W. N. Pazzi, and A. Boukerche, "Monitoring Patients Via a Secure and Mobile Healthcare System," *IEEE Wireless Communications,* vol. 17, pp. 59-65, February 2010. doi: 10.1109/MWC.2010.5416351

[49] D. Apiletti, E. Baralis, G. Bruno, and T. Cerquitelli, "Real-Time Analysis of Physiological Data to Support Medical Applications," *IEEE Trans. Inf. Technol. Biomed.,* vol. 13, pp. 313-321, May 2009. doi: 10.1109/TITB.2008.2010702

[50] C. Liu, Q. Zhu, K. A. Holroyd, and E. K. Seng, "Status and Trends of Mobile-Health Applications for Ios Devices: A Developer's Perspective," *Journal of Systems and Software,* vol. 84, pp. 2022-2033, November 2011. doi: 10.1016/j.jss.2011.06.049

[51] Z. Lv, F. Xia, G. Wu, L. Yao, and Z. Chen, "Icare: A Mobile Health Monitoring System for the Elderly," in *Proc. 3rd IEEE/ACM Int. Conf. GreenCom and CPSCom*, Hangzhou, China, 2010, pp. 699-705. doi: 10.1109/GreenCom-CPSCom.2010.84

[52] J. V. Brocke, A. Simons, B. Niehaves, K. Riemer, R. Plattfaut, and A. Cleven, "Reconstructing the Giant: On the

Importance of Rigour in Documenting the Literature Search Process," in *Proc. 17th European Conf. Information Systems*, Verona, Italy, 2009, pp. 2206-2217.
[53] B. Downe-Wamboldt, "Content Analysis: Method, Applications, and Issues," *Health Care for Women International,* vol. 13, pp. 313-321, 1992/01/01 1992. doi: 10.1080/07399339209516006
[54] S. Elo and H. Kyngäs, "The Qualitative Content Analysis Process," *Journal of Advanced Nursing,* vol. 62, pp. 107-115, November 2008. doi: 10.1111/j.1365-2648.2007.04569.x
[55] S. Easterbrook, J. Singer, M. Storey, and D. Damian, "Selecting Empirical Methods for Software Engineering Research," in *Guide to Advanced Empirical Software Engineering*, F. Shull, J. Singer, and D. I. K. Sjøberg, Eds., ed Secaucus, NJ: Springer New York, 2007, pp. 285-311.
[56] J. Webster and R. T. Watson, "Analyzing the Past to Prepare for the Future: Writing a Literature Review," *MIS Quarterly,* vol. 26, pp. xiii-xxiii, June 2002.
[57] D. S. Hamu, "Enterprise Frameworks," in *Building Application Frameworks: Object-Oriented Foundations of Framework Design*, M. Fayad, D. C. Schmidt, and R. E. Johnson, Eds., ed New York, NY: Wiley, 1999, pp. 83-86.
[58] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan*, et al.*, "A Survey of Context Modelling and Reasoning Techniques," *Pervasive and Mobile Computing,* vol. 6, pp. 161-180, March 2010. doi: 10.1016/j.pmcj.2009.06.002
[59] J. R. Hoyos, J. García-Molina, and J. A. Botía, "Mlcontext: A Context-Modeling Language for Context-Aware Systems," *Electronic Communications of the EASST,* vol. 28, p. 14, 2010.
[60] R. M. Gardner and M. M. Shabot, "Patient-Monitoring Systems," in *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*, E. H. Shortliffe and J. J. Cimino, Eds., 3rd ed New York, NY: Springer, 2006, pp. 585-625.
[61] J. E. Bardram and H. B. Christensen, "Pervasive Computing Support for Hospitals: An Overview of the Activity-Based Computing Project," *IEEE Pervasive Computing,* vol. 6, pp. 44-51, January-March 2007. doi: 10.1109/MPRV.2007.19
[62] A. Krajnc and M. Hericko, "Classification of Object-Oriented Frameworks," in *Proc. IEEE Region 8 Conf. EUROCON Computer as a Tool*, Ljubljana, Slovenia, 2003, pp. 57-61 vol.2. doi: 10.1109/EURCON.2003.1248147
[63] M. O'Grady, C. Muldoon, M. Dragone, R. Tynan, and G. O'Hare, "Towards Evolutionary Ambient Assisted Living Systems," *Journal of Ambient Intelligence and Humanized Computing,* vol. 1, pp. 15-29, December 2010. doi: 10.1007/s12652-009-0003-5
[64] W. Pree, "Essential Framework Design Patterns," *Object Magazine,* vol. 7, pp. 34-37, 1997.
[65] R. E. Johnson and B. Foote, "Designing Reusable Classes," *Journal of Object-Oriented Programming,* vol. 1, pp. 22-35, June/July 1988.
[66] N. Roy, G. Pallapa, and S. K. Das, "An Ontology-Driven Ambiguous Contexts Mediation Framework for Smart Healthcare Applications," in *Proc. 1st Int. Conf. Pervasive Technologies Related to Assistive Environments*, Athens, Greece, 2008, pp. 1-8. doi: 10.1145/1389586.1389589
[67] O. Aziz, B. Lo, A. Darzi, and G. Yang, "Introduction to Body Sensor Networks," in *Body Sensor Networks*, G. Yang, Ed., ed New York, NY: Springer, 2006, pp. 1-39.
[68] World Health Organization (WHO), *The World Health Report 2008 : Primary Health Care Now More Than Ever*. Geneva, Switzerland: World Health Organization, 2008.
[69] C. Orwat, A. Graefe, and T. Faulwasser, "Towards Pervasive Computing in Health Care–a Literature Review," *BMC Medical Informatics and Decision Making,* vol. 8, pp. 26-45, June 2008. doi: 10.1186/1472-6947-8-26
[70] F. Meneses and A. Moreira, "Technology Enablers for Context-Aware Healthcare Applications," in *Mobile Health Solutions for Biomedical Applications*, P. Olla and J. Tan, Eds., ed Hershey, PA: Information Science Reference, 2009, pp. 260-269.
[71] P. Kulkarni and Y. Ozturk, "Mphasis: Mobile Patient Healthcare and Sensor Information System," *Journal of Network and Computer Applications,* vol. 34, pp. 402-417, January 2011. doi: 10.1016/j.jnca.2010.03.030