2005

Requirements engineering for mobile systems

Goksun Ilhan
San Jose State University

Follow this and additional works at: https://scholarworks.sjsu.edu/etd_theses

Recommended Citation

Ilhan, Goksun, "Requirements engineering for mobile systems" (2005). *Master's Theses*. 2811.
DOI: <https://doi.org/10.31979/etd.s9b5-2erd>
https://scholarworks.sjsu.edu/etd_theses/2811

This Thesis is brought to you for free and open access by the Master's Theses and Graduate Research at SJSU ScholarWorks. It has been accepted for inclusion in Master's Theses by an authorized administrator of SJSU ScholarWorks. For more information, please contact scholarworks@sjsu.edu.

REQUIREMENTS ENGINEERING FOR MOBILE SYSTEMS

A Thesis

Presented to

The Faculty of Computer Engineering

San Jose State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Goksun Ilhan

May 2005

UMI Number: 1431293

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI Microform 1431293

Copyright 2005 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

© 2005

Goksun Ilhan

ALL RIGHTS RESERVED

APPROVED FOR THE DEPARTMENT OF COMPUTER
ENGINEERING

Dr. Weider Yu, Thesis Advisor

Dr. Jerry Gao

Dr. Simon Shim

APPROVED FOR THE UNIVERSITY

ABSTRACT

REQUIREMENTS ENGINEERING FOR MOBILE SYSTEMS

by Goksun Ilhan

This thesis addressed the topic of requirements engineering for mobile systems. Although there are many different well-established approaches to requirements engineering, in the area of mobile systems, not many methods are introduced. This study established a requirements engineering process specifically for mobile systems. A context-based framework was established based on another framework called MObile Work Across Heterogeneous Systems (MOWAHS). Both frameworks were explained in detail before they were applied to a case study. A mobile system for problem-based learning was used in order to derive functional and non-functional requirements and to show how to use the framework. Derived requirements then were compared with the expected requirements to evaluate the context-based framework.

ACKNOWLEDGEMENTS

I would like to thank my thesis advisor Prof. Weider Yu for his contributions and guidance through every step of this thesis.

I would also like to thank my family, especially my mom, for always encouraging me to finish my graduate studies. Without her, I would have quitted a long time ago.

Finally, I would like to thank my husband for his endless support and love.

TABLE OF CONTENTS

LIST OF FIGURES	vii
LIST OF TABLES	viii
1.0 INTRODUCTION	1
1.1 Goals and Expected Outcomes of the Study.....	2
1.2 Organization of the Study	3
2.0 TECHNOLOGY REVIEW.....	6
2.1 Mobility.....	6
2.2 Characteristics of Mobile Systems.....	9
2.3 The Requirements Engineering (RE) Process.....	11
2.4 Challenges in Requirements Engineering for Mobile Systems	14
2.5 Context Aware Computing	16
2.6 RE Methods for Mobile Systems	18
3.0 MOWAHS CHARACTERIZATION FRAMEWORK.....	22
3.1 General Characteristics	24
3.2 Information Characteristics.....	26
3.3 Location Characteristics.....	27
3.4 Time Characteristics	28
3.5 Requirement Indicators.....	32
3.5.1 Directly Computed Indicators.....	33
3.5.2 Derived Indicators.....	34
4.0 PROPOSED CONTEXT-BASED FRAMEWORK.....	37
4.1 Spatio-Temporal Context.....	42
4.2 Personal Context	47
4.3 Task Context	47
4.4 General Context	50
4.5 Environmental Context	54
4.6 Social Context.....	54
4.7 Requirement Indicators.....	57
4.7.1 Direct Indicators.....	58
4.7.2 Derived Indicators.....	61
4.8 Scalability	62
5.0 CASE STUDY	64
5.1 Problem-Based Learning (PBL)	64
5.2 Context-Based Framework Applied to PBL	69
5.3 MOWAHS Framework Applied to PBL.....	84
6.0 FINDINGS AND RESULTS.....	88
7.0 CONCLUSION AND RECOMMENDATIONS	91
References.....	94
Appendix A. Context-based Framework Application Tables for PBL Tasks	99
Appendix B. Context-based Framework Software	123

LIST OF FIGURES

Figure 1. Re-conceptualization of mobility	9
Figure 2. The RE process	14
Figure 3. Context hierarchy	17
Figure 4. MOWAHS characterization framework	23
Figure 5. Proposed context-based framework	38
Figure 6. Process steps for context-based framework	39
Figure 7. Comparison of requirement indicators (radar chart)	79
Figure 8. Comparison of requirement indicators (line-column chart)	80
Figure 9. Comparison of characteristics	80
Figure 10. Comparison of characteristics (cont.)	81
Figure 11. Comparison of characteristics (cont.)	81
Figure 12. Comparison of characteristics (cont.)	82
Figure 13. Comparison of characteristics (cont.)	82
Figure 14. Comparison of characteristics (cont.)	83
Figure 15. Comparison of characteristics (cont.)	83
Figure 16. Comparison of characteristics (cont.)	84
Figure 17. Comparison of context-based and MOWAHS frameworks	90
Figure B1. Table relationships	127
Figure B2. Main page	128
Figure B3. Create project page	129
Figure B4. Project saved page	130
Figure B5. Create tasks page	131
Figure B6. Tasks saved page	132
Figure B7. Create system target profile page	133
Figure B8. Add actual characteristics values page	134
Figure B9. Calculated system characteristics values	135
Figure B10. Calculated system attributes values	136

LIST OF TABLES

Table 1. Three dimensions of mobility and those extended perspectives.....	8
Table 2. Selected studies in mobile systems.....	20
Table 3. MOWAHS process steps	24
Table 4. Summary of MOWAHS characteristics	30
Table 5. MOWAHS summary of the indicators	33
Table 6. Rules for indicator of requirement factors.....	40
Table 7. Rules for indicator of requirement attributes.....	40
Table 9. Summary of context-based framework characteristics.....	55
Table 12. System target profile for PBL.....	70
Table 13. System target profile for PBL (cont.)	71
Table 14. Summary of the context-based framework expected characteristics vs. actual values	73
Table 15. Summary of the expected indicators of requirement attributes vs. actual values	75
Table 16. Summary of the system target profile.....	76
Table 17. MOWAHS framework applied to PBL tasks	84
Table 18. Summary of the MOWAHS framework characteristics.....	86
Table 19. Summary of MOWAHS requirement indicators	87
Table A1. Context-based framework applied to T1.....	99
Table A2. Context-based framework applied to T2.....	101
Table A3. Context-based framework applied to T3.....	103
Table A4. Context-based framework applied to T4.....	105
Table A5. Context-based framework applied to T5.....	107
Table A6. Context-based framework applied to T6.....	109
Table A7. Context-based framework applied to T7.....	111
Table A8. Context-based framework applied to T8.....	113
Table A9. Context-based framework applied to T9.....	115
Table A10. Context-based framework applied to T10.....	117
Table A11. Context-based framework applied to T11.....	119
Table A12. Context-based framework applied to T12.....	121
Table B1. Context table	123
Table B2. Attribute table.....	123
Table B3. Characteristics table	124
Table B4. System table	124
Table B5. CharSetNo table	124
Table B6. Task table	125
Table B7. ActualCharValues table	125
Table B8. SystemTargetProfileContext table	125
Table B9. SystemTargetProfileAttribute table	126
Table B10. SystemTargetProfileCharacteristics table	126

1.0 INTRODUCTION

In today's everyday life, with the need for being on the move most of the time, mobile systems have become more important. People need to access information at any time, any place, under any circumstance. This dynamic need for information access forces the technology to lean towards the mobility. Mobility is defined as the quality of moving freely. In technology, mobility defines people moving around and accessing information, services, and resources through wireless communication. As a result of this need, new hardware, software, and services are being developed for mobile systems.

Although there are many new technological developments, the current mobile systems are still far away from being ideal. There can be many reasons behind this problem. However, believing that effective software engineering methods are the starting point to design, develop, and maintain quality systems, this thesis focuses only on the requirements engineering (RE) step of the system development life cycle for mobile systems. During this thesis, mobile system is used in substitution for mobile software system.

As indicated earlier, this study focuses only on the requirements engineering process for mobile systems. The reason behind this decision is the need for a well-defined and general requirements engineering process for mobile systems. Although there are many requirements engineering methods defined for traditional systems, not many of them are applicable to mobile systems because of the differences between the traditional and mobile systems.

For this study, a process model for requirements engineering for mobile systems is proposed. Based on this process model, a context-based framework for mobile systems is developed. This context-based framework is based on another framework called MOBILE Work Across Heterogeneous Systems (MOWAHS). MOWAHS framework is explained in detail in Section 3.0. Both context-based and MOWAHS frameworks are applied to an example mobile system in order to identify the functional and non-functional requirements of the mobile system. Then, the requirements are compared to each other in order to identify the differences between the two frameworks.

1.1 Goals and Expected Outcomes of the Study

The main goal of this study is to define a requirements engineering process for the general application to all mobile systems. Since there are many differences between traditional and mobile systems, a requirements engineering process defined for traditional systems cannot be easily applied to mobile systems. Because of this, a specific process for mobile systems is highly anticipated. This process should be defined in such a manner that the difficulties concerning the requirements elicitation, analysis, specification, and validation will be eliminated or minimized.

The secondary goal is to develop a framework that defines the different characteristics describing the mobile systems. This framework is used during the requirements engineering process to classify the system characteristics under different contexts. Contexts are used to interpret the conditions of a participant involved in a mobile system. By doing so, they make the interaction between the participant and the

mobile system more efficient. Contexts are also used to utilize the information related to the condition of a participant so that the appropriate services are provided. Therefore, the proposed framework should be context-based and developed using the context hierarchy. Different contexts related to mobile computing are explained in Section 2.4.

The third goal of this study is the successful application of the proposed requirements engineering process and the context-based framework to a mobile system. This part of the study is used for evaluating the proposed requirements engineering method for general use for the mobile systems.

The expected outcomes of this study are:

1. The requirements engineering process that is used for requirements elicitation, analysis, specification, and validation for mobile systems. This process should be applicable to all mobile systems.
2. The context-based framework that is used as part of the requirements engineering process to characterize and prioritize the different contexts and characteristics closely related to mobile systems. Context-based framework should also be applicable to all mobile systems.
3. The context-based framework software. Refer to Appendix B for software application of the framework.

1.2 Organization of the Study

The organization of this study is categorized into six sections. The first section is the introduction to the intended study, goals, and expectations.

The second section focuses on the technological review of the current state of the mobile systems and its characteristics as well as the requirements engineering process, methods, and challenges. Context-aware computing is explained briefly since the proposed framework is context-based.

The third section describes the MOWAHS framework that the proposed framework is based on. Location, time, task, and general characteristics used in MOWAHS framework are explained in detail. Requirement indicators that are directly computed and derived from the framework characteristics are also explained.

Section four describes the proposed solution for requirements engineering process for mobile systems and context-based framework. Process steps are explained in detail. Different contexts of the framework and their characteristics are identified. Direct and derived indicators for requirement attributes and factors for different contexts are explained.

Section five is used to demonstrate how the proposed context-based framework and MOWAHS framework are applied to a problem-based learning (PBL) mobile system. PBL is explained briefly before both frameworks are applied and requirement indicators are derived. In the context-based framework, the actual values derived from the requirements engineering process are compared to the expected values to identify the problem areas in requirements.

Section six summarizes the findings and results of this study. Requirement indicators derived with context-based framework are compared with the indicators from

the MOWAHS framework. This comparison is followed by the comparison of both frameworks and their requirements engineering processes.

Section seven is the conclusion of the study and the recommendations for future research.

2.0 TECHNOLOGY REVIEW

For the technology review section of this study, conference papers, articles, and studies that are closely related to the requirements engineering for mobile systems are examined. Technical aspects of the mobile systems that are related to this study are introduced.

There are many different methods and approaches to the requirements engineering process. However, taken into consideration of the differences of mobile systems, these methods become very challenging to apply to mobile systems. The following sections summarize the characteristics of mobile systems, the requirements engineering (RE) process, characteristics and difficulties of requirements engineering for mobile systems, and context-aware computing.

2.1 Mobility

In order to understand mobile systems, the meaning of mobility must be defined first. According to Krogstie, mobility is defined as:

Mobility is primarily about *people* moving around, having wireless access to information and services. Secondly, mobility relates to all sorts of things in the environment with the possibility to interact with other devices and in this way influence the flow of work of the people interacting with these devices. (Krogstie, 2001)

Kakihara and Sorensen define mobility as “not just a matter of people traveling but, far more importantly, related to the *interaction* they perform - the way in which they interact with each other in their social lives.” (2002) They suggest that looking at three different types of human interaction can expand the concept of mobility. The three types

of interaction they focus on are spatial, temporal, and contextual.

Spatial mobility refers “not only to extensive geographical movement of people; it also signifies the global flux of objects, symbols, and space itself.” (Kakihara, Sorensen, 2002) In their paper, Kakihara and Sorensen state that although corporeal geographical freedom is the most recognized way of mobility, there are other aspects to be taken into consideration. These aspects are objects, symbols, and space. The objects are the mobile devices people carry around such as mobile phones or laptops. Symbols represent broadcast images and sound. Finally space represents the “virtual community or cyber community.” (Kakihara, Sorensen, 2002)

Temporal mobility investigates the concerns about the question 'when'. It takes into consideration the changes in technology and how it affects the usage of time and planning of activities (Churchill, Munro, 2001). Based on Hall's studies, (1959, 1962) Kakihara and Sorensen distinguish between the monochronicity and polychronicity temporalities. Monochronicity refers to people planning and allocating specific time slots for their activities. Simply, it means doing one specific task at a time. Polychronicity refers to multitasking, people performing several tasks at a time.

Contextual mobility “refers to the situation and environment in which humans perform their activities. More specifically, contextuality provides an understanding in what way and circumstance the activity is being performed.” (Basole, 2004) According to Kakihara and Sorensen, the context in which the human interaction occurs is as important as spatiality and temporality. “That is, interactional aspects such as ‘in what

way,’ ‘in what particular circumstance,’ and ‘towards which actor(s)’ the action is performed constitute the crucial disposition of interaction just as the aspects ‘where’ and ‘when’ do.” (Kakihara, Sorensen, 2002)

In Table 1, three dimensions of mobility are summarized from the perspective of Kakihara and Sorensen.

Table 1. Three dimensions of mobility and those extended perspectives

Source: Kakihara, M., Sorensen, C. (2002). Mobility: An Extended Perspective.

Retrieved March 1, 2005, from

http://www.kakihara.org/papers/Kakihara&Sorensen_HICSS35.pdf

Dimensions of mobility	Aspects of interaction	Extended perspectives
Spatiality	- Where	Geographical movement of not just human but objects, symbols, images, voice, etc.
Temporality	- When	- Clock time vs. Social time * Objective vs. Subjective - Monochronicity vs. Polychronicity
Contextuality	- In what way - In what circumstance - Towards which actor(s)	- Multi modality of interaction - Unobtrusive vs. Obtrusive - Ephemeral vs. Persistent - Weakly & strongly tied social networks

Similarly, Pica, Sorensen, and Allen define three kinds of mobility: locational, operational, and interactional. (2004) Locational mobility refers to the “extensive geographical movement with the mobile device.” (Pica, Sorensen, Allen 2004)

Operational mobility refers to the “flexible coordination of daily operations through the mobile device.” (Pica, Sorensen, Allen, 2004) Finally, interactional mobility refers to the “intense interaction with diverse people and data through the mobile device.” (Pica, Sorensen, Allen, 2004) Adapted from Kakihara, Figure 1 shows the re-conceptualization

of mobility (Pica, Sorensen, Allen, 2004).

Figure 1. Re-conceptualization of mobility

Source: Pica, D., Sorensen, C., Allen, D. (2004). On Mobility and Context of Work: Exploring Mobile Police Work. Retrieved September 4, 2005, from <http://is.lse.ac.uk/staff/sorensen/download/PicaSorensenAllen.pdf>

2.2 Characteristics of Mobile Systems

Mobile systems are different from traditional systems in many ways. Comparing mobile systems to traditional systems, the following characteristics of the mobile systems are found to be noticeably different from traditional systems: (Shen, Shen 2001; Krogstie, 2001)

- The interaction between the system and the user is dynamic. Users should be able to use the system at any time, any location, and under any condition. Information and services should be available regardless of the environment the user is in. Therefore the interaction has to be accomplished regardless of the different contexts.
- Hardware is not standardized and mobile devices are usually used for single-purpose

or limited operations. This causes the mobile devices to have weaker hardware components (memory, storage, etc.). Another reason is the size of the hardware devices. They are expected to be portable which limits the hardware configuration.

- Human interaction with the system is harder because of the small input and output devices. Because mobile devices are expected to be portable, the input (e.g., keypad) and the output (e.g., screen) interfaces are limited in size with respect to the mobile device.
- Physical system boundaries are undefined or vaguely defined. This affects the security of the system. As a result of this, new security aspects such as identification/authentication, authorization/access control, data integrity, confidentiality, and non-repudiation should be taken into consideration.
- There is no standard for defining user interfaces. User interfaces can be different for different mobile devices and systems.
- Users need the mobile system for immediate usage. There is no typical usage scenario for the mobile system. There can be hundreds of mobile scenarios that make it almost impossible to design a well-defined mobile system.
- Applications are highly related and dependent to the context that the users are in. Different contexts affect the mobile systems in different ways. In order to cover the different effects on the system, mobile system should be designed as context-aware.
- Users can only pay limited attention to mobile applications because of the distractions surrounding them. Since mobile systems can be used anywhere, anytime, users are

faced with different kinds of distractions such as visual or auditory. Another reason for this is the involvement of the user in multiple activities.

- Activities happen in spurts. Users are involved in multiple activities while using the mobile system. Users may want to use the mobile system for different activities at the same time.
- Energy consumption is a big concern since mobile devices are portable and have limited battery power.
- Data and communication transparency is very important because of the different mobile devices and different communication networks involved in mobile computing. If transparency is not provided, mobile system will have limited usage and this defies the purpose of mobile systems.

All of the above explain the characteristics of mobile systems that are different from traditional systems. These characteristics make it very hard to design and implement an efficient and traditional mobile system.

2.3 The Requirements Engineering (RE) Process

The first step in building a software system is the identification of system requirements to run the system. As Fred Brooks indicated:

The hardest part of building a software system is deciding precisely what to build. No other part of the conceptual work is so difficult as establishing the detailed technical requirements... No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later. (Brooks, 1987)

Requirements engineering can be defined as the iterative process of gathering, analyzing, specifying, validating, managing, and documenting conditions and capabilities

needed to solve a problem or achieve a goal.

The RE process consists of four steps:

- 1. Requirements elicitation:** Requirements elicitation is the activity of discovering and driving requirements from stakeholders and system requirements. Requirements elicitation begins with identification of all stakeholders. Stakeholders are the users, customers, experts, and developers of the system. The purpose of requirements elicitation is to find the facts, establish a common frame of reference, and resolve open issues. (Brackett, 1990) Common frame of reference is established when all involved stakeholders agree on the meaning of the gathered information. Information can be gathered by using surveys, questionnaires, interviews, and observations. In order to understand the problem and the need, relevant questions are asked and gathered information is checked for consistency.
- 2. Requirements analysis:** Requirements analysis is the activity of checking information for conflicts, inconsistencies, and ambiguities. According to Brackett, there are three steps in requirements analysis phase. (1990) These steps have to be repeated until all the issues, conflicts, inconsistencies, and ambiguities are resolved.

The first step is the assessment of potential problems. During this step, gathered information is checked for ambiguities, inconsistencies, and incompleteness. Requirements are investigated to find out if they are feasible to apply. System requirements are compared with the derived requirements.

The second step is the classification of requirements. During this step,

requirements are classified into groups and then they are prioritized. Brackett suggests prioritizing requirements into three categories. These categories are mandatory, desirable, and inessential. Mandatory requirements are the most important requirements of the system and they have to be implemented. A change in a mandatory requirement affects the whole system. Desirable requirements have a lower priority. However, they need to be implemented as well. Although changes in this type of requirements affect the system, they do not have as strong impact as mandatory requirements. Inessential requirements do not have to be implemented unless the schedule allows it. They do not affect the system very much.

The third step is the evaluation of feasibility and risks. During this step, requirements are assessed for technical, operational, and economic feasibilities. Technical feasibility focuses on the possibility of building the system depending on the gathered requirements (can the system be built with the current technology?). Operational feasibility focuses on the possibility of operating the system. Economic feasibility focuses on the cost of building and maintaining the system.

3. **Requirements specification:** Requirements specification is the activity of representing the information gathered for establishing communication and understanding between the stakeholders of the system. This representation is usually in the form of documents. Requirement specification document is used to resolve inconsistencies and open issues. This document also provides a way to trace requirements back to their identification step.

4. **Requirements validation:** Requirements validation is the activity of inspecting the requirements once again for conflicts, inconsistencies, ambiguities, and quality. Involved stakeholders must agree on the techniques to be used to develop the system and the criteria for the final acceptance. This step provides a common frame of reference between the stakeholders so that there will be no problems during and after the implementation of the system.

The RE process described above is summarized in Figure 2:

Figure 2. The RE process

Source: McPhee, C., Eberlein, A. (2002). Requirements Engineering for Time-to-Market Projects. Retrieved September 17, 2003, from <http://csdl.computer.org/dl/proceedings/ecbs/2002/1549/00/15490017.pdf>

2.4 Challenges in Requirements Engineering for Mobile Systems

Requirements engineering is the essential starting point of software development process. “Without a well-defined requirements engineering (RE) process, the resulting system is likely to reveal wrong features, not meet market needs, be out of schedule, and

bug-infested.” (Maccari, 1999) Therefore, it is no exception for mobile systems to try to follow a well-defined RE process. However, because of the differences between mobile systems and traditional systems, there are many challenges in establishing a well-defined RE process. The following section explains the challenges in the RE process for mobile systems.

First of all, there are no specific methods developed for collecting and analyzing the need for a new service for mobile systems. Mostly, users of the mobile systems do not know what they want or need from a new mobile system. Users have the ability to discuss and opionate about an existing service. However, when it comes to a new service, it is very hard to acknowledge the need and establish requirements from that specific need.

Secondly, there are no fixed forms of interaction between the users and the mobile system. In the development process for traditional systems, it is necessary to identify the typical scenarios and use cases. This is very hard for mobile systems since there are many possible scenarios. The usage of the system is also dependant on the context. Even if the analysis of the requirements is done correctly, requirement specification becomes very hard to accomplish with the traditional requirements engineering process.

Thirdly, requirements elicitation and evaluation steps of the RE process require different methods for different users, aspects, and environments. Since mobile systems need to adapt to dynamic environments and contexts, one specific method is not sufficient for elicitation and evaluation of the requirements in mobile systems.

Lastly, because of the competitive nature of mobile and wireless systems in today's market, there is not enough time for requirements gathering activities. These activities are time consuming. Therefore the requirements engineering process is not fully applied to the current mobile systems. Only limited requirements engineering can be accomplished within strict project schedules.

All these challenges make it very hard to apply traditional requirements engineering methods to mobile systems to the full extent. This study takes this problem as a starting point to establish a different requirements engineering process specific to mobile systems.

2.5 Context Aware Computing

As mentioned before, the interaction between the mobile system and the user is complex and dynamic. Users should be able to use the system to access information under any condition. According to Dey and Abowd, context is defined as:

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. (Dey and Abowd, 1999)

Therefore, context can be described as any information that describes the condition of a partaker in an interaction in the system. Moran and Dourish (2001) describe context as referring to the situation, both physical and social, that computational devices are embedded in. One of the goals of the context aware computing is to gather and utilize the information related to the context of the device so that the appropriate services are provided. Figure 3 shows the hierarchy of different contexts in relation to the mobile user.

Figure 3. Context hierarchy

Source: Kofod-Petersen, A., Aamodt, A. (2003). Case-Based Situation Assessment in a Mobile Context-Aware System. Retrieved March 15, 2004, from <http://w5.cs.uni-sb.de/~krueger/aims2003/camera-ready/kofod-petersen-6.pdf>

The context hierarchy model is divided into the following five categories:

- **Task context:** Task context describes what the user is doing. It identifies what tasks the user is involved in and what the user's goals are.

The task context can be described with explicit goals, tasks, actions, activities, or events. Notice that this also can include other people's tasks (that are within the situation). (Goker, Myrhaug, 2002)

- **Social context:** Social context describes the social aspects of the user. It can contain information about co-workers, friends, neighbors, and relatives.

One important aspect in a social context is the role that the user plays in the context. A role can be described with a name, the user's status in this role, the tasks that the user can perform in this role, and the various sub-roles that the role can have. A role can in addition be played in a social arena. (Goker, Myrhaug, 2002)

- **Personal context:** This context consists of two subparts: the physiological context and the mental context. The physiological context can contain information like pulse, blood pressure, weight, retinal pattern etc. The mental context can contain information like mood, expertise, angeriness, and stress etc.

- **Spatio-temporal context:** This context describes aspects of the user relating to the time and space.

It can contain attributes like: time, location, direction, speed, shape (of objects/buildings/terrain), track, place, clothes of the user and so on. i.e., the spatial extension of the environment and the things in it. (Goker, Myrhaug, 2002)

- **Environmental context:** This context describes the entities that surround the user.

These entities can be things, services, temperature, light, humidity, noise, and people. Information (e.g., text, images, movies, sounds) which is accessed by the user in the current user context is all part of the environment context. The various networks that are in the surrounding can also be described in the user's environment context. (Goker, Myrhaug, 2002)

2.6 RE Methods for Mobile Systems

As indicated before, there are many challenges in the RE process for mobile systems. These challenges prevent to establish and define a well-established RE process for mobile systems. In order to improve the process of requirements engineering, the following approaches for mobile systems are proposed by Shen, Shen (2001):

- **Context based requirements gathering:** Mobile systems are highly dependent on different contexts. Different contexts cause users and systems to act differently.

Therefore, it is a necessity to examine users' everyday lives and try to identify the context-based tasks and their implications on the system. This method then can be used for identifying and establishing the requirements for the mobile system.

- **Prototyping:** Prototyping can be used in order to educate users and developers.

Although it is a time consuming task to create a prototype of the mobile system, it can be very resourceful for establishing user interface requirements that are not clearly

defined in the iterative development process.

- **Formative evaluation:** It is a method proposed by Galal (2000) for the iterative development process. The main idea is to develop a prototype and improve it according to the user feedback. By this method, qualitative information is transferred into quantitative information. Analyzing the quantitative information can provide useful information about the requirements.
- **Scenario analysis:** Although there are many different mobile scenarios and use cases in mobile systems, the most common scenarios should be identified for this method. These scenarios provide important information on the general usage of the mobile system. They are also used to describe the user needs and goals.
- **User profile establishment:** A user profile is established by using the characteristics in the user context. Although it is an impossible task to categorize all the users of the system, it is very important to establish a user profile for a specific group of users. The nature of the mobile system market is changing because of the competition. However, a specific group of users can help to identify the need for a new product and its requirements.

The studies on requirements engineering for mobile systems usually use formative evaluation, prototyping, or user profile establishment as their methods. Table 2 shows some of the selected studies of RE for mobile systems. All these studies focus on gathering user requirements. Not one of these studies tries to derive system requirements. Once again this shows the strong need for a well-established RE process

for mobile systems. In this thesis, context-based requirements gathering method is used to formulate the RE process specific to mobile systems.

Table 2. Selected studies in mobile systems

Source: Shen X., Shen J. (2001). User Requirements in Mobile Systems. Retrieved October 27, 2003, from <http://web.njit.edu/~jxs1866/papers/amcis01-requirements.PDF>

Researchers	Purpose	Subjects	Methods	Simple Findings
Pascoe et. al.	To get usability of mobile devices in field work environment	Fieldworkers e.g., ecologists, archeologists	Iterative process using Prototyping applications on Palm Pilots	<ul style="list-style-type: none"> • Needs for in-directive operation, e.g., eye-free / single-hand operations • Needs for context-awareness applications, e.g., use a GPS receiver attached to Palm and automatically fill in location info
Brown et. al.	To learn Information capture practice at work for design of information capture devices	11 professionals who worked at Hewlett Packard	<ul style="list-style-type: none"> • Diary method using digital cameras for one week to record incidents of information capturing • Semi-structured interviews 	<ul style="list-style-type: none"> • Taxonomy of information capture activities • Detailed descriptions of subjects' capture practice.
Koskinen	Short Message Service—Its current use, impact on social norms, values etc.	Young adults between 20-25 years	<ul style="list-style-type: none"> • Ethnographic research, iterative process • Individual/group interview • Participatory observation 	(Expected results will be used for user interface & application design for Nokia)

Väänänen-Vainio-Mattila & Ruusk	Gather Users' needs and design ideas for Nokia 9000	Communicator 6 mobile professionals	<ul style="list-style-type: none"> • Observation for a few hours of each subject. • Contextual inquiry and affinity diagram to identify emerging themes of users' activities. 	<ul style="list-style-type: none"> • Users needed to organize info according to their own logic—provide means for personalization • Frequent contact info needed to be treated differently than other contacts.
Palen et. al.	Behavior and practice of new mobile phone users	19 new mobile phone users	<ul style="list-style-type: none"> • Three interviews for each user during the first 6 weeks after service acquisition • Voice-mail “diaries” 	Mobile devices were integrated into daily life and social life.
Hjelmeroo et. al. (Usability study)	Nokia 9000 Communicator browser usability	<p>Group1: novices/friendly Users</p> <p>Group 2: error hunters + developers of third-party software</p>	<ul style="list-style-type: none"> • Interview (video taped) • Questionnaire 	<ul style="list-style-type: none"> • “Hot list” as default home page worked fine • Functional layers were confusing • “Back” & “Previous” buttons were confusing
Nielsen J. (Usability study)	Effectiveness of WAP (Wireless Application Protocol)	20 experienced mobile phone users in UK	<ul style="list-style-type: none"> • Self-recorded diary • Usability test • Interview 	<ul style="list-style-type: none"> • 70% said that they would not be using WAP in a year • Time-consuming and expensive. e.g., check local weather took about 2 minutes

3.0 MOWAHS CHARACTERIZATION FRAMEWORK

As indicated earlier, the purpose of this study is to create a framework that is used to identify and elicit both functional and non-functional requirements of a mobile system. Such a framework is created as a result of this study based on another framework called MOBILE Work Across Heterogeneous Systems (MOWAHS). In order to compare and contrast these two frameworks, MOWAHS framework is explained before the proposed solution. Although the wording is changed, all information on MOWAHS framework is taken from Sorensen et al.'s "The MOWAHS Characterization Framework for Mobile Work" and Ramampirao et al.'s "Requirement Indicators for Mobile Work: The MOWAHS Approach" papers.

MOWAHS is a research project sponsored by Norwegian Research Council. MOWAHS was first initiated in January 2001. The goal of MOWAHS framework is described as:

The aim of the MOWAHS characterization framework is to produce requirements for an end-system supporting mobile computing. Such an end-system can be a process support system for mobile work consisting of a server or a set of servers, some mobile clients (laptops, PDAs, mobile phones etc.), and human resources required to carry out a work process. The characteristics of the framework will give requirements to the architecture (topology, connectivity, network, type of client(s), type of server(s), reliability, security, performance, etc.), to services needed, and requirements for hardware to be used. (Sorensen et al., 2002)

The MOWAHS framework describes the tasks of a scenario of a mobile system by using four main categories of characteristics shown in Figure 4.

Figure 4. MOWAHS characterization framework

Source: H., Ramampiaro, Sorensen, C., Wang, A., Nam Le, H., Nygard, M., and Conradi, R. (2003). Requirement Indicators for Mobile Work: The MOWAHS Approach. Retrieved February 16, 2004, from <http://www.idi.ntnu.no/grupper/su/publ/pdf/ai2002-mowahs-indicator.pdf>

The focus of the framework is on different tasks and roles. Numerical scale of 1-5 is used to describe different characteristics of a specific task. High complexity of system requirements is represented by high values whereas lower values represent lower complexity in requirements. The process of applying the framework is described step by step in Table 3.

Table 3. MOWAHS process steps

Source: Sorensen, C., Wang, A., Nam Le, H., Ramampiaro, H., Nygard, M., and Conradi, R. (2002). The MOWAHS Characterization Framework for Mobile Work. Retrieved February 16, 2004, from <http://www.idi.ntnu.no/grupper/su/publ/pdf/ai2002-mowahs-framework.pdf>

1. Select the mobile scenario
2. Identify different roles in the scenario
3. For each role; identify tasks and responsibilities
4. For each task:
 - a) Assign task weight
 - b) Characterize the task using the characterization framework
5. Apply the process characteristics:
 - a) Calculate weighted mean values for each characteristic
 - b) Calculate an overall scenario complexity indicator
6. Derive system requirements and priorities from the process characteristics

In order to understand how MOWAHS framework applies to a mobile scenario, characteristics of the framework are explained in detail in the following section.

3.1 General Characteristics

“The general characteristics are used to describe task structures and attributes that are indirectly related to mobility. The words in parenthesis describe the possible answers and measurements of each characteristic element.” (Sorensen et al., 2002) General characteristics are used to determine the general task indicator (GTI).

G1 Decomposable (1 None, 3 Uncertain, 5 Yes)

G1 is used to describe the task structure. It is a characteristic used to decide if a task can be decomposed into smaller, sub-tasks. “This can be used to decide if the end-system needs to support hierarchical composition of tasks.” (Sorensen et al., 2002)

G2 Part of sequence (1 No, 3 Partial, 5 Yes)

G2 is used to determine if a task has dependencies of order to other tasks. “This will decide if the end system needs specification of execution order.” (Sorensen et al., 2002)

G3 Pre-planned (1 Planned, 3 Partial, 5 Ad-hoc)

G3 is used to determine if a task is planned in beforehand. This can be used to provide the communication requirements, back-end support, and support for task planning and creation. “If a task is pre-planned, information required for a specific task can be provided in beforehand on a mobile device.” (Sorensen et al., 2002) G3 is also used to determine the network connectivity indicator (NCI) and transactional support indicator (TSI).

G4 Data synchronization (1 No, 3 After task, 5 Within task)

G4 is used to determine the time interval in which the updated data is synchronized with the other tasks. This can be used to provide the communication requirements. G4 is also used to determine the network connectivity indicator (NCI) and transactional support indicator (TSI).

G5 Data exchange rate (1 None, 3 Once, 5 Many)

G5 is used to determine the exchange rate of data between tasks. This can be used to determine the network capacity and system complexity. G5 is also used to determine the network connectivity indicator (NCI), network speed indicator (NSI), energy consumption indicator (ECI), and transactional support indicator (TSI).

3.2 Information Characteristics

The information characteristics are used to describe the information required and/or produced, information type, and information transfer speed. These characteristics are used to determine the network capacity of the mobile system as well as information complexity indicator (ICI).

I1 Information contents (1 NA, 2 Text, 3 Graphics, 4 Audio, 5 Video)

I1 is used to determine the type of information needed/required by the task. It specifies the media that the system must support. I1 is used to establish the Quality of Service (QoS) requirement. I1 is also used to compute the value for energy consumption indicator (ECI).

I2 Information streaming (1 NA, 3 Discrete, 5 Continuous)

I2 describes the type of data streaming. I2 is used to determine the information exchange and mobile host devices. I2 also affects the Quality of Service (QoS) and network capacity. I2 is used to compute the energy consumption indicator (ECI).

I3 Information required related to time (1 NA, 2 Low, 3 Medium, 4 High, 5 Real-time)

I3 describes the type of information required by the task. In other words, I3 describes how important the information required to execute the task is related to time. I3 is used to compute the network speed indicator (NSI).

I4 Information produced related to time (1 NA, 2 Low, 3 Medium, 4 High, 5 Real-time)

I4 describes the type of information produced by the task. I4 also describes how important the information produced by the task related to time. I4 is used to compute the network speed indicator (NSI).

I5 Information transmission speed (1 NA, 2 Slow, 3 Medium, 4 Fast, 5 Very fast)

I5 describes the transmission speed of information expected/required by the task. I5 is used to determine the network speed indicator (NSI) as well as information complexity indicator (ICI).

3.3 Location Characteristics

“The location characteristics are used to specify if a task or back-end system support depend on geographical locations.” (Sorensen et al., 2002) Location characteristics are used to determine the location complexity indicator (LCI). The following items are used to describe location characteristics:

L1 Location dependent (1 No, 3 Partial, 5 Yes)

L1 is used to determine the dependency of a task execution to a specific location. L1 can be used to decide on the services and physical equipment needed/required for task execution. This characteristic is used to compute the energy consumption indicator (ECI) as well as mobility indicator (MI).

L2 Require services at location (1 No, 3 Partial, 5 Yes)

L2 is used to determine if a task requires services available at specific location

during its execution. L2 is used to compute energy consumption indicator (ECI) as well as location complexity indicator (LCI).

L3 Produce services at location (1 No, 3 Partial, 5 Yes)

L3 is used to determine if a task produces services at specific location during its execution. These services can be used by the other tasks. L3 is used to compute location complexity indicator (LCI), energy consumption indicator (ECI), and mobility indicator (MI).

L4 Location report (1 No, 3 Partial, 5 Yes)

L4 is used to determine if a task is required to report its location to the mobile system during its execution. L4 is used to compute network connectivity indicator (NCI), energy consumption indicator (ECI), and mobility indicator (MI).

L5 Route constraints (1 No, 3 Partial, 5 Yes)

L5 is used to determine if a task is required to follow a specific network route during its execution. L5 is used to compute the mobility indicator (MI).

3.4 Time Characteristics

The time characteristics are used to determine the temporal properties of a task and the system support for these properties.

The creation and execution of tasks may require timely coordination between an end-system and the task executor. These properties (excluding task lifetime) can be used to determine the required connectivity (communication) between the end-system and mobile clients. All the characteristics may imply requirements for extended transaction support. (Sorensen et al., 2002)

T1 Event-triggered (1 No, 3 Partial, 5 Yes)

T1 is used to determine if a task is triggered by an event or not. This characteristic can be used to identify the need for instant access to information and services. T1 is used to determine the time complexity indicator (TCI), network connectivity indicator (NCI), and transactional support indicator (TSI).

T2 Time constrained (1 No, 3 Partial, 5 Yes)

T2 is used to describe if a task must be executed at a specific time or within a specific time span. “This property can be used to decide how the system should be synchronized. A time constrained task may also demand support for rescheduling of tasks and if a task is achievable or not, related to time constraints.” (Sorensen et al., 2002)

T3 Temporal coordination (1 No, 3 Partial, 5 Yes)

T3 is used to determine if a task is required/needed to coordinate with other tasks. T3 is used to compute time complexity indicator (TCI) as well as mobility indicator (MI).

T4 Task resumption (1 No, 3 Partial, 5 Yes)

T4 is used to determine if a task can halt its execution and resume where it has left off without the need for restarting the task. T4 is used to compute transactional support indicator (TSI).

T5 Task lifetime (1 Seconds, 2 Minutes, 3 Hours, 4 Days, 5 Weeks)

T5 is used to describe the expected lifetime of a task. The scale for each mobile system can be adjusted. For some systems short may mean minutes whereas for others it may mean days. The task lifetime property is used to determine the network connectivity

and the type of transaction support needed “(e.g., long, nested transactions with a need for relaxed Atomicity, Consistency, Isolation and Durability (ACID) properties.)”

Sorensen et al., 2002)

The characteristics of MOWAHS framework are summarized in Table 4.

Table 4. Summary of MOWAHS characteristics

Source: H., Ramampiaro, Sorensen, C., Wang, A., Nam Le, H., Nygard, M., and Conradi, R. (2003). Requirement Indicators for Mobile Work: The MOWAHS Approach. Retrieved February 16, 2004, from <http://www.idi.ntnu.no/grupper/su/publ/pdf/ai2002-mowahs-indicator.pdf>

Characteristic	Possible Values	Description
G1 Decomposable	1 No, 3 Uncertain, 5 Yes	Decides whether the task is composed of sub tasks or not.
G2 Part of sequence	1 No, 3 Partial, 5 Yes	Specifies if a task has order dependencies with respect to other tasks.
G3 Pre-planned	1 Planned, 3 Partial, 5 Ad-hoc	Describes to what degree a task is planned in beforehand.
G4 Data synchronization	1 No, 3 After task, 5 Within task	Specifies when a task has to synchronize/merge updated data with other tasks.
G5 Data exchange rate	1 None, 3 Once, 5 Many	Specifies the rate of data exchange between the current task and other tasks (during its lifetime).
I1 Information contents	1 NA, 2 Text, 3 Graphics, 4 Audio, 5 Video	Describes the complexity of the information required or produced by the task.
I2 Information streaming	1 NA, 3 Discrete, 5 Continuous	Describes whether the task requires streaming of data or not.
I3 Information required related to time	1 NA, 2 Low, 3 Medium, 4 High, 5 Real-time	Describes how important the information required to execute the task is related to time.

I4 Information produced related to time	1 NA, 2 Low, 3 Medium, 4 High, 5 Real-time	Describes how important the information produced by the task is related to time.
I5 Information transmission speed	1 NA, 2 Slow, 3 Medium, 4 Fast, 5 Very fast	Describing the expected transmission speed of information used or produced by the task.
L1 Location dependent	1 No, 3 Partial, 5 Yes	Describes to what degree a task must be executed at a specific location (e.g., that a lecturer must be in a classroom to teach students).
L2 Require services at location	1 No, 3 Partial, 5 Yes	Specifies if a task needs electronic services available at the location such as printers, network connections, video projectors, fax machines, etc.
L3 Produce services at location	1 No, 3 Partial, 5 Yes	Specifies if a task produces electronic services at the location that can be used by others at this location, such as information queries, beaming of information, network connectivity etc.
L4 Location report	1 No, 3 Partial, 5 Yes	Specifies if a task must report its location to the system.
L5 Route constraints	1 No, 3 Partial, 5 Yes	Specifies if a mobile task must follow a specific route or not when moving around.
T1 Event-triggered	1 No, 3 Partial, 5 Yes	Decides whether a task is triggered by an event or not.
T2 Time constraint	1 No, 3 Partial, 5 Yes	Describes if a task must be executed at a specific time or within a specific time-span (e.g., a task must be performed between 10:15 and 12:00).

T3 Temporal coordination	1 No, 3 Partial, 5 Yes	Describes if a task must be coordinated with other tasks (e.g., different military units must strike at the same time).
T4 Task resumption	1 No, 3 Partial, 5 Yes	Describes if a task can halt, and then later resume from where it left off (not requiring a complete restart of task).
T5 Task lifetime	1 Seconds, 2 Minutes, 3 Hours, 4 Days, 5 Weeks	Describes the expected lifetime of a task.

3.5 Requirement Indicators

In MOWAHS framework, different indicators are computed in order to gather and to elicit functional and non-functional requirements. There are two groups of indicators. The first group, directly computed indicators, is established by using the four groups of characteristics from the framework. The second group, derived indicators, is established by combining different characteristics from different groups of the framework. Summary of all indicators is shown in Table 5. (Ramampiaro et al., 2003)

Table 5. MOWAHS summary of the indicators

Source: H., Ramampiaro, Sorensen, C., Wang, A., Nam Le, H., Nygard, M., and Conradi, R. (2003). Requirement Indicators for Mobile Work: The MOWAHS Approach. Retrieved February 16, 2004, from <http://www.idi.ntnu.no/grupper/su/publ/pdf/ai2002-mowahs-indicator.pdf>

Indicator	Key Characteristics	Additional Characteristics
General Task Indicator (GTI)	G1-G5	None
Information Complexity Indicator (ICI)	I1-I5	None
Location Complexity Indicator (LCI)	L1-L5	None
Time Complexity Indicator (TCI)	T1-T5	None
Network Connectivity Indicator (NCI)	G3-G5, L4, T1	T4
Network Speed Indicator (NSI)	I3-I5	G5, I2
Energy Consumption Indicator (ECI)	L1-L3, T5	G5, I1, I2, L4
Transactional Support Indicator (TSI)	G3-G5, T1, T4, T5	None
Mobility Indicator (MI)	L4, L5	L1, L3, T3

3.5.1 Directly Computed Indicators

Directly computed indicators are characteristics from the four groups of MOWAHS framework.

General Task Indicator (GTI)

GTI is computed by taking the average of values G1-G5. GTI score describes the task structure and attributes. A high score represents the high complexity of the task and correspondingly the complexity of the functionality to handle the task.

Information Complexity Indicator (ICI)

ICI is computed by taking the average of values I1-I5. "A high ICI score indicates that the end-system must cope with complex information presentation, management, and transmission." (Ramampiaro et al., 2003) ICI helps to determine the Quality of Service (QoS).

Location Complexity Indicator (LCI)

LCI is computed by taking the average of values L1-L5.

A high LCI score indicates that the end system should possibly include GIS-support and should put high requirements on the type of equipment that can be used as mobile clients. In addition, it tells how the location affects the actual task performance. It also determines the required portability of the relevant mobile devices – i.e., the weight, the size, etc. A high score also indicates that some GIS-support should be supported to define the location of the task and the need for representing location in the process modeling language. Finally, it can determine the need for the system to be aware of location. This means that some data may be relevant at a specific location, other data may not, etc. (Ramampiaro et al., 2003)

Time Complexity Indicator (TCI)

TCI is computed by taking the average of values T1-T5. TCI is used to determine the non-functional system performance and system availability requirements. It is also used to determine the functional task scheduling, synchronization, and coordination requirements.

3.5.2 Derived Indicators

Derived indicators are combined characteristics from the four groups of MOWAHS framework.

Network Connectivity Indicator (NCI)

NCI is computed by taking the average of values G3, G4, G5, L4, and T1. T4 can also be used in order to drive this value. NCI is used to determine the network capabilities of the mobile system. It also helps to determine non-functional requirements of latency and reliability.

Network Speed Indicator (NSI)

NSI is computed by taking the average of values I3, I4 and I5. G5 and I2 can also be used as additional characteristics. NSI is used to determine the network speed of the mobile system. It also helps to determine non-functional requirements of latency, Quality of Service (QoS), and performance.

Energy Consumption Indicator (ECI)

ECI is computed by taking the average of values L1-L3 and T5. G5, I1, I2, and L4 can also be used.

A high ECI score means that the mobile device requires supplies that are able to deliver high energy, while a low score signifies low energy consumption. Based on this, the ECI indicates the energy technology needed that should be provided or selected, considering portability versus energy cost. (Ramampiaro et al., 2003)

Transactional Support Indicator (TSI)

TSI is computed by taking the average of values G3, G4, G5, T1, T4, and T5. “A high TSI score indicates that the transactional support must be flexible and advanced; while a low score may indicate that traditional transactional support (ACID transactions) is sufficient.” (Ramampiaro et al., 2003) The ACID concept is described in ISO/IEC 10026-1:1992 Section 4. ACID represents the four properties of transactions that are Atomicity, Consistency, Isolation and Durability. “They are considered to be the key transaction processing features of a database management system (DBMS). Without them, the integrity of the database cannot be guaranteed.” (Wikipedia, 2004) Atomicity is the “all or nothing” property. A transaction is either completed without any interruptions or is rolled back to its previous state if an error occurs. Consistency is the

property to ensure that the database is in a valid state when the transaction starts and ends. Isolation is the property to keep a transaction isolated from all the other operations while it is not completed. Durability is the ability to recover the committed data even if there is system or storage failure.

Mobility Indicator (MI)

MI is computed by taking the average of values L1, L3, L4, L5, and T3. “A high average score of the above characteristics indicates that a task is highly mobile, while a low score tells us that a task may not be mobile or has low mobility.” (Ramampiaro et al., 2003)

4.0 PROPOSED CONTEXT-BASED FRAMEWORK

The purpose of the context-based framework is to gather functional and non-functional requirements for a mobile system. This framework focuses on tasks and roles of the possible mobile scenarios as well. However, the main difference from the MOWAHS framework is that proposed framework takes contexts into consideration. Attributes of the mobile system are taken into consideration based on the characteristics of multiple contexts. Characteristics of the proposed framework are shown in Figure 5.

Figure 5. Proposed context-based framework

Figure 6 shows the process steps for context-based framework.

1. Select the mobile scenario
2. Create System Target Profile
3. Identify all tasks of mobile scenario
4. For each task:
 - a) Assign task weight
 - b) Assign actual values for each characteristic using the characterization framework
5. For system:
 - a) Calculate weighted mean values for each characteristic
 - b) Calculate requirement indicators
6. Derive system requirements and priorities from the framework characteristics and requirement indicators
7. Compare derived system requirements with System Target Profile

Figure 6. Process steps for context-based framework

The first step is to choose the mobile scenario for the system. Mobile scenario represents the actual flow of events of the system. “Scenarios specify how users carry out their tasks in a specified context. They provide examples of usage as an input to design.” (Usability Net, 2005)

The second step is to create the system target profile. System target profile lists the expected values for each context, requirement indicator, and system characteristics. Starting at the characteristics level, for each characteristic, an expected value for the whole system is assigned. To give an example, expected value for characteristic STT1 Scheduled, can be the values of 1 (not scheduled), 3 (partially scheduled), or 5 (scheduled). This indicates that if the system consists of scheduled tasks or not as a whole. Going up a level, for each requirement indicator another expected value is assigned. To continue the example, after assigning expected values to STT1, STT2,

STT3, STT4, and STT5, time indicator (STI) is given an expected value. STI indicates all time requirements for the mobile system. The last level is the context level. For each context, an expected value of Satisfied or Not Satisfied is assigned. This value indicates the level of satisfaction each context is expected to have. If a value of Satisfied is assigned to spatio-temporal context, actual, calculated requirement indicators for time and location attributes have to match the expected values. For the comparison of actual and expected values of indicators of requirement factors, the rules in Table 6 have to be followed.

Table 6. Rules for indicator of requirement factors

Indicator of Requirement Attribute	Indicator of Requirement Factor
	Not Satisfied
Almost	Not Satisfied
	Satisfied

For the values of indicator of requirement attributes, the rules in Table 7 have to be followed. $A\alpha(i)$ represents the expected value for attribute i. $A\beta(i)$ represents the actual value of i.

Table 7. Rules for indicator of requirement attributes

Condition	Indicator of Requirement Attribute
If the difference between the expected and actual values of attributes is > 0.2 ($A\alpha(i) - A\beta(i) > 0.2$)	
If the difference between the expected and actual values of attributes is ≤ 0.2 ($A\alpha(i) - A\beta(i) \leq 0.2$)	Almost
If the difference between the expected and actual values of attributes is $= 0$ ($A\alpha(i) - A\beta(i) = 0$)	

After the creation of the system target profile, each task in the scenario is identified and assigned a weight depending on the importance of that task for the whole system. “Task analysis analyzes what a user is required to do in terms of actions and/or cognitive processes to achieve a task. A detailed task analysis can be conducted to understand the current system and the information flows within it.” (Usability Net, 2005) Then, each task is assigned its actual value for each characteristic.

The next step is to calculate the actual characteristic values for system level. Actual values are computed using Equation 1. This is the weighted mean of all tasks. $C\beta(i)$ represents the actual value of characteristic for each task. w_i is the task weight and w_j is the total weight of all tasks. n and m represent the number of tasks.

$$\sum_{i=1}^n \frac{w_i * C\beta(i)}{\sum_{j=1}^m w_j} \quad (1)$$

Actual system level characteristic values are applied to Equation 2 to calculate the indicators of requirement attributes. $C\alpha(i)$ represents the expected value for characteristics at system level. $C\beta(i)$ represents the actual value for characteristics at system level. w_i represents the ratio of characteristics expected value to the total expected value.

$$\text{Indicator of Requirement Attribute} = A\alpha(i) * \left(\sum_{i=1}^n w_i * \chi_i \right) \quad (2)$$

$$w_i = \frac{C\alpha(i)}{\sum_{i=1}^n C\alpha(i)}$$

$$\chi_i = \begin{cases} \frac{C\alpha(i)}{C\alpha(i)} = 1 \\ \frac{C\beta(i)}{C\alpha(i)} \end{cases}$$

Table 8 shows which part of the equation to be used based on different conditions.

Table 8. Conditions for indicator of requirement attributes calculation

Condition	Replace formula with
If Expected value for characteristic ($C\alpha(i)$) is less than or equal to Actual value ($C\beta(i)$) $C\alpha(i) \leq C\beta(i)$	$\chi_i = \begin{cases} \frac{C\alpha(i)}{C\alpha(i)} = 1 \end{cases}$
If expected value for characteristic ($C\alpha(i)$) is greater than Actual value ($C\beta(i)$) $C\alpha(i) > C\beta(i)$	$\chi_i = \begin{cases} \frac{C\beta(i)}{C\alpha(i)} \end{cases}$

The last step is to compare the actual requirement indicators with expected ones. If the two values match, it is an indication that the requirements are correct and no changes need to be made. If the two values do not match, each characteristic has to be checked to see which one causes the problem. The characteristic that causes the problem is the area where system requirements have to be changed. System characteristic has to be traced back to the task characteristic that initiated the problem. After changing the value, process steps 4b-7 has to be implemented again.

In order to understand the context-based framework, contexts, their attributes, and characteristics are explained in detail in the following section.

4.1 Spatio-Temporal Context

Although spatio-temporal context contains attributes like time, location, direction, speed, and place, context-based framework focuses on the time and location attributes of

the task in the mobile scenario. Spatio-temporal context is used to determine the requirements for location based information and services, and real-time processing.

STT1 Scheduled (0 N/A, 1 No, 3 Partial, 5 Yes)

According to Online Dictionary of Computing, task scheduling is defined as “the assignment of start and end times to a set of tasks, subject to certain constraints. Constraints are typically either time constraints (the payload must be installed before the payload bay doors are closed) or resource constraints.” (Online Dictionary of Computing, 1998) STT1 specifies if a task is planned/scheduled to happen at a specific time. STT1 can be used to decide how the task should be synchronized with respect to the other tasks. STT1 is used to determine the spatio-temporal time indicator (STI). For example, Professor X giving a lecture on February 28, 2004 at 3:00 pm in Engineering Building, Room 322 is a scheduled task. However, the task of students listening to Professor X’s lecture is partially scheduled since they do not have to be present in the room or attend the presentation. Students can obtain the notes or audio recordings of the lecture from another student at a later time.

STT2 Event-triggered (0 N/A, 1 No, 3 Partial, 5 Yes)

STT2 specifies if a task is caused by another event or not. Event can be defined as an occurrence that is predefined and detected by the system. STT2 is used to determine the need for available services. STT2 is used to determine the spatio-temporal time indicator (STI). To give an example, let’s say that every week Professor X has a class on Monday at 3:00pm in Room 322. Thirty minutes before the class, student Y goes to

Room 322 and sets up the projector and calls Professor X telling him that the classroom is ready. Afterwards, Professor X goes to the class to give his lecture. In this case, the Professor X's going to the class is an event-triggered task. It is based on Student Y calling him. Let's say this week Student A sets up the projector but forgets to call Professor X. Although he did not receive a call, Professor X assumes that Student A set up the projector and he goes to give his lecture. In this case, the task is partially event-triggered.

STT3 Information flow type (0 N/A, 1 None, 3 One-way, 5 Two-way)

STT3 specifies the type of the information flow. There may be one-way, two-way, or no information flows between the system and the user of the system. One-way information flow can specify a reply/feedback to the user from the system or it can be an input from the user to the system. Two-way information flow specifies an interactive communication between the user and the system. STT3 is used to compute network capacity indicator (NCI), mobility indicator (MI), and spatio-temporal time indicator (STI). For example, Professor X giving the lecture contains one-way information flow whereas students asking questions and Professor X answering these questions contain two-way information flow.

STT4 Information flow (0 N/A, 1 None, 2 Text, 3 Graphics, 4 Audio, 5 Video)

STT4 specifies what type of information is required/requested by the system/user. STT4 is important for the Quality of Service (QoS) of the mobile system. STT4 is used to determine the media mobile system should support. STT4 is used to compute network

capacity indicator (NCI), mobility indicator (MI), and spatio-temporal time indicator (STI). To continue with the previous example, students can learn about Professor X's lecture by reading the lecture notes (text), looking at the presentation slides (graphics), listening to audio recording (audio), or watching the video recording (video) of the lecture.

STT5 Task lifetime (0 N/A, 1 Short, 3 Medium, 5 Long)

STT5 specifies the task lifetime, how long a task needs to run in order to accomplish its duty. Running time of the task is determined in relation to the running time of other tasks. There is no specific time interval, such as minutes, hours, etc., that determines the category of the characteristic. For example, preparing the presentation for the lecture is a long task. Listening to the presentation is a medium task and talking about the presentation is a relatively short task. A reference point can be constructed based on the whole system. If a task takes more than 20% of the whole system, this task can be categorized long. Likewise, if a task is more than 10% but less than 20% of the whole system, this task can be categorized as medium. Finally, if a task is less than 1% of the whole system, it can be categorized as short.

STL1 Location dependency (0 N/A, 1 No, 3 Partial, 5 Yes)

STL1 specifies if a task is dependent to a specific location. It describes if the user of the system is required to be at a specific place in order to accomplish the task. This characteristic is used to determine the complexity of the mobile system. STL1 is used to determine the spatio-temporal location indicator (SLI), network capacity indicator (NCI),

and mobility indicator (MI). For example, let's say that Professor X presents his lecture in Engineering Building room 322. For his students to be able to listen to the actual presentation, they have to be present in room 322. In this case, listening to Professor X's lecture is a location dependant task. However, if the lecture is broadcasted in multiple locations on campus, students do not have to be present in room 322, but they have to be on campus. In this case, listening to the lecture is a partially-dependant task.

STL2 Location report (0 N/A, 1 No, 3 Partial, 5 Yes)

STL2 specifies if a task is required/requested to report its location to the system. STL2 is used in connection with STL3 so that a task can receive services available at the specific location. For example, after listening to Professor X's lecture, student A wants to download and then print the notes on campus. He can search for the nearest printer by sending a search request through his mobile device and the list of printers can be sent back to his mobile device. After choosing the closest printer, he can send lecture notes to the printer. In this case, student A's mobile device should send a location report to find the closest printer.

STL3 Service availability (0 N/A, 1 No, 3 Partial, 5 Yes)

STL3 specifies if a task is required/requested to know about the services available at the specific location. Continuing with the above example, let's say that student A enters a room with multiple printers and he is not sure which printer to use. A message indicating which printers are available in the room can be sent to his mobile device.

4.2 Personal Context

Personal context focuses on the expertise attribute of the user in the mobile scenario. Expertise attribute is part of the mental context. Context-based framework does not include the physiological context attributes. Personal context is used to determine the usability requirements of the system.

PE1 User experience required (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

PE1 specifies what level of user experience is required so that the user is able to use the mobile system. PE1 is used to determine the user interface of the system.

4.3 Task Context

Task context focuses on the structure, behavior, and complexity attributes of the tasks in the mobile scenario.

TS1 Decomposable (0 N/A, 1 No, 3 Partial, 5 Yes)

TS1 specifies if a task is composed of other smaller tasks. TS1 is used to determine the task structure indicator (TSI). For example, Professor X giving the lecture consists of multiple smaller tasks such as, preparing the lecture material, creating the presentation files, arranging an available room, arranging a projector, etc.

TS2 Sequential (0 N/A, 1 No, 3 Partial, 5 Yes)

TS2 specifies if a task needs to be handled in sequence with respect to the other tasks. This characteristic is used to determine the execution order of system tasks. TS2 is also used to compute the task structure indicator (TSI). For example, for students to ask questions to Professor X about the presentation, the presentation task has to be

completed.

TS3 Interdependent (0 N/A, 1 No, 3 Partial, 5 Yes)

TS3 specifies if a task has dependencies to other tasks. This characteristic is used to determine the task dependencies and their order of execution. It also specifies the coordination and cooperation between tasks. For example in order for Professor X to give the lecture, he has to prepare the presentation files first. Therefore, actual presentation task is interdependent with preparing the presentation task. There is an execution order for those two tasks.

TS4 Collaboration type (0 N/A, 1 No, 2 Client/Server, 3 Peer-to-Peer, 4 Broadcasting, 5 Ad-hoc)

TS4 specifies the type of collaboration between different tasks executors. Client/Server collaboration is a computing network in which the functions are divided between clients and servers. Server provides resources for clients to use. Peer-to-peer collaboration is a computing network that treats all computers as equal. Individual computers can share resources. There is no server and each computer can provide services. Broadcasting is the delivery of data from one computer to all computers in the computer network. According to searchMobileComputing web page “an ad-hoc (or ‘spontaneous’) network is a local area network or other small network, especially one with wireless or temporary plug-in connections, in which some of the network devices are part of the network only for the duration of a communications session or, in the case of mobile or portable devices, while in some close proximity to the rest of the network.”

(searchMobileComputing, 2003)

TB1 Repetitiveness (0 N/A, 1 No, 3 Partial, 5 Yes)

TB1 specifies if a task is needed/required to be executed more than once. TB1 is used to determine the task behavior indicator (TBI). Let's say after a popular request, Professor X decides to repeat his presentation one more time, this task becomes repetitive. However, if Professor X decides that he has not enough time to repeat his presentation but he goes over the important parts one more time, this task becomes partially repetitive.

TB2 Task completion/failure notification (0 N/A, 1 No, 3 Partial, 5 Yes)

TB2 specifies if a notification is required/requested upon completion/failure of the task. TB2 is used to determine the task behavior indicator (TBI). For example, after his presentation, Professor X requires his students to write a paper and send it to him via e-mail. If he only replies to e-mails indicating that he received the papers, this task has partial task completion/failure notification. If he sends e-mails to students who did not send their papers saying that he did not receive their papers, then this task has full task completion/failure notification.

TC1 Complexity (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

TC1 specifies the complexity of the task. TC1 is used to determine the task complexity indicator (TCI). For example listening to Professor X's lecture has low complexity. However, if students are required to write a paper about the presentation, this task will have medium complexity since they have to research about the subject. If

students are required to write a paper and take a test after the presentation, this task will have high complexity.

4.4 General Context

General context focuses on the attributes that are not covered by other five contexts. These attributes are security, scalability, transparency, availability, and efficiency.

GSec1 Identification/ Authentication (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GSec1 specifies the ability/need to mutually identify and authenticate communicating parties. This characteristic is used to determine the system security requirements. It is used to compute general security indicator (GSI). For example, if Professor X only allows students who are enrolled in his class to listen to his presentation then this task has high identification/authentication requirements. If he lets anyone from the Computer Engineering Department to be able to listen to his presentation then this task requires medium identification/authentication. If the presentation is open to San Jose State University students, then this task requires low identification/authentication. If he lets anyone in the presentation then there are no identification/authentication requirements.

GSec2 Authorization/Access Control (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GSec2 specifies the level of authorization/access control required by the task. Authorization and access control means that only properly authenticated persons or communicating entities receive permission to access security-critical data, applications, or other resources. For example if Professor X puts his presentation material on his web

page and gives out passwords to download it, then this task has high access control. If he lets anyone with a student id to reach the materials, then this task has medium access control. If he does not require passwords but he puts the material on the intranet, only accessible if someone is a San Jose State University student, then it has low access control requirement.

GSec3 Data Integrity (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GSec3 specifies the level of data integrity required by the task. Data integrity provides an assurance that data received after a successful transmission is authentic.

GSec4 Confidentiality (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GSec4 specifies the level of confidentiality required by the task. Information sent over some communication channel cannot be understood by anyone for whom it was unintended. For example, let's say that Professor X wants to make an announcement about his class. If he puts the information on his web page then there is no confidentiality. If he sends e-mail to all of his students in understandable text then the confidentiality is low. If he sends e-mail to all his students but encrypts the message then the confidentiality is medium. Only his students have the key to decrypt the message. If he sends separate e-mails to each student and encrypts the message and each student has to get separate keys then the confidentiality is high.

GSec5 Non-repudiation (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GSec5 specifies the level of non-repudiation required by the task. The beneficiary has evidence regarding from whom the data or application comes or has some evidence

that an intended receiver actually received data send to it.

GSca1 Number of nodes (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GSca1 specifies the number of nodes in the network. This characteristic is used to determine the network capacity and scalability of the mobile system. When number of nodes is high, the scalability of the system is also high.

GSca2 Density (0 N/A, 1 No, 2 High, 3 Medium, 5 Low)

GSca2 specifies the density of the network. It is the number of nodes per square kilometer. This characteristic is also used to determine the network capacity and scalability of the mobile system. When density increases, the scalability of the system decreases.

GSca3 Traffic (0 N/A, 1 No, 2 High, 3 Medium, 5 Low)

GSca3 specifies the traffic level of the network. When traffic of the network increases, the scalability of the system decreases.

GT1 Data transparency (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GT1 specifies the level of data transparency required by the task. Data transparency provides sharing of data between sending and receiving parties regardless of how the data is represented. Data can be encoded at the sending end. However, when the receiving end gets it, system should be able to translate it to its original form. GT1 is used to determine the transparency of the mobile system.

GT2 Communication transparency (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GT2 specifies the level of communication transparency required by the task.

Communication transparency provides transparent communication between different tasks. GT2 is also used to determine the transparency of the whole system.

GA1 Computer down time (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GA1 specifies the ability of the task to run when the computer is down. GA1 is used to determine the availability of the system.

GA2 Communication network down time (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GA2 specifies the ability of the task to run when the communication network is down. GA2 is also used to determine the availability of the system.

GA3 Communication network not available (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GA3 specifies the ability of the task to run when the communication network is not available.

GE1 Cost to run (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GE1 specifies the cost the user of the system has to spend in order to run the task. This cost can be hardware, software, and subscribed services.

GE2 Cost to setup (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GE2 specifies the cost the user has to spend for setup of the system. This cost may include subscribing to services, setting up the software on hardware, and setup time.

GE3 Cost to maintain (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

GE3 specifies the cost to maintain the system. Maintenance may include software/hardware updates, subscription renewals, and keeping the system operable.

GE1, GE2 and GE3 are used to determine the requirement for efficiency of the system.

As the cost increases, the efficiency of the system decreases.

4.5 Environmental Context

Environmental context focuses on the physical and social attributes of the user and his/her surroundings. For context-based framework, visual/auditory distraction and interaction characteristics are taken into consideration.

EP1 Visual distraction (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

EP1 specifies the level of visual distraction the user is involved in during the execution of the task.

EP2 Auditory distraction (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

EP2 specifies the level of auditory distraction the user is involved in during the execution of the task.

ES1 Interaction (0 N/A, 1 No, 2 Low, 3 Medium, 5 High)

ES1 specifies the level of interaction required/needed for task execution.

4.6 Social Context

Social context focuses on the social aspects of the user while executing the task. For context-based framework, role type and role priority are taken into consideration.

SR1 Role type (0 N/A, 1 No, 3 Individual, 4 Facilitator, 5 Group)

SR1 specifies the role of the user during the task execution. User can be the initiator of the task as well as the receiver.

SR2 Role priority (0 N/A, 1 No, 3 Secondary, 5 Primary)

SR2 specifies the priority of the role during the task execution.

The characteristics, attributes and possible values of the proposed context-based framework are summarized in Table 9.

Table 9. Summary of context-based framework characteristics

Context	Attribute	Characteristic	Values
Spatio-temporal	Time	STT1: Scheduled	0 N/A, 1 No, 3 Partial, 5 Yes
		STT2: Event-triggered	0 N/A, 1 No, 3 Partial, 5 Yes
		STT3: Information flow type	0 N/A, 1 None, 3 One-way, 5 Two-way
		STT4: Information flow	0 N/A, 1 None, 2 Text, 3 Graphics, 4 Audio, 5 Video
		STT5: Task lifetime	0 N/A, 1 Short, 3 Medium, 5 Long
	Location	STL1: Location dependency	0 N/A, 1 No, 3 Partial, 5 Yes
		STL2: Location report	0 N/A, 1 No, 3 Partial, 5 Yes
		STL3: Service availability	0 N/A, 1 No, 3 Partial, 5 Yes
Personal	Expertise	PE1: User experience required	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
Task	Structure	TS1: Decomposable	0 N/A, 1 No, 3 Partial, 5 Yes
		TS2: Sequential	0 N/A, 1 No, 3 Partial, 5 Yes
		TS3: Interdependent	0 N/A, 1 No, 3 Partial, 5 Yes
		TS4: Collaboration type	0 N/A, 1 No, 2 Client/Server, 3 Peer-to-Peer, 4 Broadcasting, 5 Ad-hoc
	Behavior	TB1: Repetitiveness	0 N/A, 1 No, 3 Partial, 5 Yes
		TB2: Task completion/failure notification	0 N/A, 1 No, 3 Partial, 5 Yes
	Complexity	TC1: Complexity	0 N/A, 1 No, 2 Low, 3 Medium, 5 High

General	Security	GSec1: Identification/Authenti- cation	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GSec2: Authorization/Access Control	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GSec3: Data Integrity	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GSec4: Confidentiality	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GSec5: Non- repudiation	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
	Scalability	GScal1: Number of nodes	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GScal2: Density	0 N/A, 1 No, 2 High, 3 Medium, 5 Low
		GScal3: Traffic	0 N/A, 1 No, 2 High, 3 Medium, 5 Low
	Transparency	GT1: Data transparency	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GT2: Communication transparency	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
	Availability	GA1: Computer down time	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GA2: Comm. Network down time	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GA3: Comm. Network not available	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
	Efficiency	GE1: Cost to run	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GE2: Cost to setup	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		GE3: Cost to maintain	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
Environmental	Physical	EP1: Visual distraction	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
		EP2: Auditory distraction	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
	Social	ES1: Interaction	0 N/A, 1 No, 2 Low, 3 Medium, 5 High
Social	Role	SR1: Role type	0 N/A, 1 No, 3 Individual, 4 Facilitator, 5 Group

		SR2: Role priority	0 N/A, 1 No, 3 Secondary, 5 Primary
--	--	--------------------	-------------------------------------

4.7 Requirement Indicators

In context-based framework, different indicators are also computed in order to specify requirements. Like in MOWAHS framework, there are two types of requirement indicators. The first group of indicators is established directly by using the six context characteristics from the framework. The second group is the combination of different characteristics from different contexts. Numerical scale of 1-5 is used to describe different requirement indicators of a mobile system. High complexity of system requirements is represented by high values. Lower values represent lower complexity in requirements. Summary of all indicators is shown in Table 10.

Table 10. Requirement indicators

Indicator Type	Indicator Name	Characteristics
Direct Indicators	Spatio-temporal Time Indicator (STI)	STT1-STT5
	Spatio-temporal Location Indicator (SLI)	STL1-STL3
	Personal Expertise Indicator (PEI)	PE1
	Task Structure Indicator (TSI)	TS1-TS4
	Task Behavior Indicator (TBI)	TB1-TB2
	Task Complexity Indicator (TCI)	TC1
	General Security Indicator (GSI)	GSec1-GSec5
	General Scalability Indicator (GScaI)	GSca1-GSca3
	General Transparency Indicator (GTI)	GT1-GT2
	General Availability Indicator (GAI)	GA1-GA3

	General Efficiency Indicator (GEI)	GE1-GE3
	Environmental Physical Indicator (EPI)	EP1-EP2
	Environmental Social Indicator (ESI)	ES1
	Social Role Indicator (SRI)	SR1-SR2
Derived Indicators	Network Capacity Indicator (NCI)	STT3, STT4, STL1, STL2, STL3, TB2, GSca1, GSca2, GSca3, GT1 and GT2
	Mobility Indicator (MI)	STL1, STL2, STT3 and STT4
	Transactional Support Indicator (TSI)	STT1-STT3, STT5 and TB2
	Energy Consumption Indicator (ECI)	STL1, STL3 and STT5

4.7.1 Direct Indicators

Spatio-temporal Time Indicator (STI) (0 N/A, 1 Low, 3 Medium, 5 High)

STI is computed by using Equation 2 for values STT1-STT5 for all tasks. STI is used to determine system performance, task scheduling, and synchronization. STI also helps determining the system complexity and Quality of Service (QoS). High STI value indicates that the system should be able to support time-critical or real-time task execution and system operation.

Spatio-temporal Location Indicator (SLI) (0 N/A, 1 Low, 3 Partial, 5 High)

SLI is computed by using Equation 2 for values STL1-STL3 for all tasks. SLI is used to determine the requirement of location awareness and location dependency. Location aware systems can use the services available to that specific location. High SLI value indicates that the system should be aware of the location and the available services.

Personal Expertise Indicator (PEI) (0 N/A, 1 Low, 3 Medium, 5 High)

PEI is used to determine the complexity of the system and user interface. High PEI value indicates that the mobile system interface is complex and user needs to be more experienced in order to use the system. Low PEI value indicates that user interface should be simple and easy to use.

Task Structure Indicator (TSI) (0 N/A, 1 Low, 3 Medium, 5 High)

TSI is computed by using Equation 2 for values TS1-TS4 for all tasks. TSI is used to determine the structure of the tasks and how to handle them. TSI indicates the complexity of the mobile system.

Task Behavior Indicator (TBI) (0 N/A, 1 Low, 3 Medium, 5 High)

TBI is computed by using Equation 2 for values TB1 and TB2 for all tasks. TBI is used to determine the behavior of the tasks and how to handle them. TBI also indicates the complexity of the mobile system as well as TSI.

Task Complexity Indicator (TCI) (0 N/A, 1 Low, 3 Medium, 5 High)

TCI is used to determine the overall complexity of the task. It helps to find out about the complexity of the whole system. A high TCI value indicates that the mobile system is complex and needs supporting system design and architecture.

General Security Indicator (GSI) (1 N/A, 2 Low, 3 Medium, 5 High)

GSI is computed by using Equation 2 for values GSec1-GSec5 for all tasks. GSI is used to determine the security requirements of the mobile system. A high GSI values indicates that the system should have a high level of security. GSI also helps to

determine the authentication, access control, identification, and data integrity requirements.

General Scalability Indicator (GScaI) (0 N/A, 1 Low, 3 Medium, 5 High)

GScaI is computed by using Equation 2 for values GSca1-GSca3 for all tasks.

GScaI is used to determine the non-functional scalability requirements of the mobile system. A high GScaI value indicates that the system should be highly scalable.

General Transparency Indicator (GTI) (0 N/A, 1 Low, 3 Medium, 5 High)

GTI is computed by using Equation 2 for values GT1 and GT2 for all tasks. GTI is used to determine the transparency requirements of the mobile system. High GTI value indicates that system should be highly transparent for both data and communication. High transparency means that regardless of what hardware is used in the network, communication has to be accomplished between different types of devices without notifying the users.

General Availability Indicator (GAI) (0 N/A, 1 Low, 3 Medium, 5 High)

GAI is computed by using Equation 2 for values GA1-GA3 for all tasks. GAI is used to determine the availability requirements of the system. High GAI value indicates that the system can be available even though the computers or the communication network are down. A high GAI value is desired for the system.

General Efficiency Indicator (GEI) (0 N/A, 1 Low, 3 Medium, 5 High)

GEI is computed by using Equation 2 for values GE1-GE3 for all tasks. GEI is used to determine the efficiency requirements of the mobile system. High GEI value

indicates that the efficiency is low. Low GEI value is desired.

Environmental Physical Indicator (EPI) (0 N/A, 1 Low, 3 Medium, 5 High)

EPI is computed by using Equation 2 for values EP1 and EP2 for all tasks. EPI is used to determine how much distraction a user is facing while using the system. EPI should be taken into consideration while designing the system and user interface of the system. A high EPI value indicates that the user is in a physically distracted environment.

Environmental Social Indicator (ESI) (0 N/A, 1 Low, 3 Medium, 5 High)

ESI is the value of ES1. A high ESI value indicates the social interaction of the user with his/her environment. ESI is used to determine the requirements of the user interface and network capacity.

Social Role Indicator (SRI) (0 N/A, 1 Low, 3 Medium, 5 High)

SRI is computed by using Equation 2 for values SR1 and SR2 for all tasks. SRI is used to determine the role of the user while executing the task and using the system.

4.7.2 Derived Indicators

Network Capacity Indicator (NCI) (0 N/A, 1 Low, 3 Medium, 5 High)

NCI specifies the connection, speed, and support requirements of the network for the mobile system. NCI is derived from STT3, STT4, STL1, STL2, STL3, TB2, GSca1, GSca2, GSca3, GT1, GT2, GA1, GA2, and GA3 by using Equation 2. Underlying architecture requirements of the network are very important depending upon the complexity of the whole system. Network should be able to provide all the

required/requested information, actions, and services.

Mobility Indicator (MI) (0 N/A, 1 Low, 3 Medium, 5 High)

MI specifies the mobility of the system. MI is derived from STL1, STL2, STT3, and STT4 by using Equation 2. A high value indicates that the task is mobile. MI determines the requirements for network technology in the mobile system. Refer to Section 2.1 for mobility.

Transactional Support Indicator (TSI) (0 N/A, 1 Low, 3 Medium, 5 High)

TSI specifies the level of transactional support needed/required by the system. TSI is derived from STT1-STT3, STT5, and TB2 by using Equation 2. A high value indicates that the mobile system needs a more relaxed and complicated ACID properties. ACID properties are explained in Section 3.5.2 Transactional Support Indicator for MOWAHS. A low value indicates the system is able to use traditional, strict ACID properties.

Energy Consumption Indicator (ECI) (0 N/A, 1 Low, 3 Medium, 5 High)

ECI specifies the level of energy consumption required by the system. ECI is derived from STL1, STL3, and STT5 by using Equation 2. A high value indicates that the mobile system requires supplies to deliver high energy. A low value indicates that the mobile system requires low energy consumption. ECI is used to determine the requirements for energy supplies and hardware.

4.8 Scalability

Scalability can be defined as the ability of a system to adapt itself to the current circumstances to perform its usual operations. One of the important aspects of context-

based characterization framework for mobile systems is its flexibility to be adapted to different mobile systems. All of the characteristics and contexts of the context-based characterization framework may not be applicable to all systems. In this case the user of the framework may choose to exclude any characteristics and/or contexts. Similarly, if the need for new/improved characteristics and/or contexts arises, the user again may adapt these new additions to the framework easily. However, the calculations for the actual values and requirement indicators have to take these changes into consideration.

5.0 CASE STUDY

It is desirable to apply the context-based framework and MOWAHS framework to a case study so that proposed framework can be evaluated. The chosen case study is mobile problem-based learning (PBL) system software. In order to understand this case study, problem-based learning is explained briefly before the application of the framework.

5.1 Problem-Based Learning (PBL)

According to Barrows and Tamblyn, "problem-based learning (PBL) is the learning which results from the process of working toward the understanding and resolution of a (complex) problem." (1980)

Initially, the class is divided into several teams. The team size is usually small, four to seven students each. Teams remain the same throughout the semester. Students are presented with a problem instead of traditional presentation of a specific topic. Teams follow the predefined steps of the PBL process towards the gaining of knowledge and skills. As described by Barrows (1988), the main goals of PBL are:

- To develop students' thinking and reasoning skills
- To develop students as independent, self-disciplined learners

According to Uden and Dix (2000), PBL helps students to:

- Engage in a problem
- Reason effectively based on presented/gained knowledge
- Achieve the desirable outcome

- Identify and acknowledge their faults in PBL process
- Work as a member of a team

Adapted from Barrows (1988), Uden and Dix (2000), Table 11 shows the four phases of PBL.

Table 11. Four phases of PBL

Phase	Activity	Explanation	Details	By Whom
Phase 1 – Problem	Present problem	What is the problem?	<ul style="list-style-type: none"> • Boundaries • Constraints • Components • Interactions • Requirements • Issues • Problem Statement 	Facilitator
	Set hypothesis	What do we know?	<ul style="list-style-type: none"> • Facts • Data • Knowledge 	Group (with facilitator)
		What solutions are possible?	<ul style="list-style-type: none"> • Causation • Effect • Possible Resolution 	Group (with facilitator)
	Identify learning	What criteria will be used to judge the solutions?	<ul style="list-style-type: none"> • Objective Measures • Subjective Measures 	Group (with facilitator)
	Assign tasks	How do we distribute the work?	<ul style="list-style-type: none"> • Develop task list • Prioritize tasks • Assign tasks to group members 	Group (with facilitator)
Phase 2 - Learning	Consult resources	What resources do we need to gather information?	<ul style="list-style-type: none"> • Information retrieval • How and where to find it • Brainstorm possible resources 	Individual and Group

	Detailed study of issues	What do we need to know?	<ul style="list-style-type: none"> • Facts • Data • Knowledge • Skills • Values • Develop questions to be answered 	Individual
Phase 3 - Solving	Evaluate resources	Is the gathered information any good? Present information gathered and discuss with the team.	<ul style="list-style-type: none"> • What have I learned? • What have my team members learned that will help me? 	Group
	Re-examine the problem	Process the data	<ul style="list-style-type: none"> • Extract relevant data • Integrate new information into the context of the problem • Organize the information • Generate possible solutions based on all of the known information 	Group
	Revise hypothesis and learning issues	Evaluate alternatives and make a decision	<ul style="list-style-type: none"> • Analyze the solutions • Agree upon one as being the best • Use selection criteria from Phase 1 	Group
Phase 4 - Reflection	Present findings	Present best solution	<ul style="list-style-type: none"> • Build a model? • Create a project or prototype? • Develop a product? 	Group

		Document the proposed solution	<ul style="list-style-type: none"> • Draft paper or proposal • Communication skills 	Group
	Metacognitive critique	Is the decision appropriate?	<ul style="list-style-type: none"> • Does it properly address the problem? • Were the procedures appropriate? • Does the decision respect the values and criteria? 	Facilitator and Peers
		What have we learned?	<ul style="list-style-type: none"> • About this problem • About the criteria and problems • About procedures • About the problem solving process • About ourselves and others • What went right during the project? • What went wrong during the project? 	Group
	Reformulate further hypotheses	What similar problems could I solve this way?		Group
		What information from your project would help subsequent teams?		Group

The following is the list of twelve tasks of PBL:

T1. Present problem: The facilitator presents the problem to the students. Boundaries, constraints, components, and requirements are explained. Problem statement is constructed.

T2. Set hypothesis: Each group, with the help of the facilitator, discusses the facts, data, and knowledge known about the problem. Then, each group formulates the hypothesis.

T3. Identify learning: Each group, with the help of the facilitator, discusses the objective and subjective measures to judge the possible solutions.

T4. Assign tasks: Each group, with the help of the facilitator, discusses how to distribute the work among the group members. Possible task lists are created and prioritized. Each group member is assigned tasks.

T5. Consult resources: Each group member, with the help of the other members, brainstorms on the possible resources. How and where to retrieve the information is discussed.

T6. Detailed study of issues: Each group member researches the information about his/her appointed tasks. Facts and data are gathered.

T7. Evaluate resources: Each group member presents the collected information to his/her group and each group discusses this information.

T8. Re-examine the problem: Each group processes the collected information and discusses the relevancy of this information. Information is organized and possible solutions are generated based on gathered information.

T9. Revise hypothesis and learning issues: Each group analyzes the possible solutions and choose one as being the best and most appropriate.

T10. Present findings: Each group documents and presents its proposed solution.

T11. Metacognitive critique: After the presentation of the proposed solutions, each solution is discussed by the facilitator and other groups in order to find out if it was appropriate for the problem or not. Methods, procedures, and problem solving process are also discussed to formulate the learning process.

T12. Reformulate further hypotheses: Each group discusses the similar problems to solve.

5.2 Context-Based Framework Applied to PBL

In order to apply context-based framework to PBL, the system target profile has to be created first. System target profile shows the expected values for each context, attribute, and characteristic. Expected values for context can be Satisfied and Not Satisfied. Satisfied value indicates that the expected values of the attributes should match calculated values. Not Satisfied value indicates that the expected and calculated values do not match. Tables 12-13 show the system target profile for PBL.

Table 12. System target profile for PBL

Indicator of Requirement Factor (Context)	Expected value for Requirement Factor	Indicator of Requirement Attribute	Expected value for Requirement Attribute	Comments
Spatio-temporal	Satisfied	Time (STI)	3	System is expected to consist of partially time triggered and scheduled tasks.
		Location (SLI)	3	System is expected to be partially location dependant.
Personal	Satisfied	Expertise (PEI)	2	User experience required is expected to be low.
Task	Satisfied	Structure (TSI)	3	System is expected be structured.
		Behavior (TBI)	3	System tasks are expected be partially repeated.
		Complexity (TCI)	3	System complexity is expected to be medium.
General	Satisfied	Security (GSI)	2	Security of the system can be low.
		Scalability (GScal)	4	System should be highly scalable.
		Transparency (GTI)	4	System should be highly transparent.
		Availability (GAI)	4	System should be highly available.
		Efficiency (GEI)	3	System efficiency can be medium.
Environmental	Satisfied	Physical (EPI)	3	Physical distraction of the user is expected to be medium.
		Social (ESI)	4	Social interaction of the user is expected to be high.
Social	Satisfied	Role (SRI)	4	Role of the user is expected to be either initiator or participant.

	Satisfied	Network Capacity (NCI)	4	Network capacity is expected to be high.
	Satisfied	Mobility (MI)	4	Mobility is expected to be high.
	Satisfied	Transactional Support (TSI)	3	Transactional support can have medium ACID properties.
	Satisfied	Energy Consumption (ECI)	3	Energy consumption is expected to be medium.

Table 13. System target profile for PBL (cont.)

Indicator of Requirement Attribute	Characteristic	Expected Value for Requirement Characteristics	Comments
Time (STI)	STT1: Scheduled	3	System is partially scheduled.
	STT2: Event-triggered	3	System is partially event-triggered.
	STT3: Information flow type	5	Flow is two-way.
	STT4: Information flow	2	Flow is at least text.
	STT5: Task lifetime	3	System lifetime is medium.
Location (SLI)	STL1: Location dependency	3	System is partially location-dependant.
	STL2: Location report	3	System partially needs location report.
	STL3: Service availability	3	Services are needed partially.
Expertise (PEI)	PE1: User experience required	2	User experience is low.
Structure (TSI)	TS1: Decomposable	3	Tasks are partially decomposable.
	TS2: Sequential	3	Tasks are partially sequential.
	TS3: Interdependent	3	Tasks are partially interdependent.

	TS4: Collaboration type	5	Collaboration is Ad-hoc.
Behavior (TBI)	TB1: Repetitiveness	3	Some system tasks are repeated.
	TB2: Task completion/failure notification	3	Notification is partially needed.
Complexity (TCI)	TC1: Complexity	3	System has medium complexity.
Security (GSI)	GSec1: Identification/Authentication	2	Identification is sometimes needed.
	GSec2: Authorization/Access Control	2	Access control is sometimes needed.
	GSec3: Data Integrity	2	Data integrity has low importance.
	GSec4: Confidentiality	2	Confidentiality is low.
	GSec5: Non-repudiation	2	Non-repudiation is low.
Scalability (GScaI)	GSca1: Number of nodes	3	# of nodes are medium.
	GSca2: Density	3	Density is medium.
	GSca3: Traffic	3	Traffic is medium.
Transparency (GTI)	GT1: Data transparency	4	Data transparency is highly needed.
	GT2: Communication transparency	4	Comm. transparency is highly needed.
Availability (GAI)	GA1: Computer down time	4	When one comp. is down, system should be available.
	GA2: Comm. Network down time	3	When network is down, system should be partially available.
	GA3: Comm. Network not available	3	When network is not accessible, system should be partially available.
Efficiency (GEI)	GE1: Cost to run	2	Cost to run is low.
	GE2: Cost to setup	3	Cost to setup is medium.

	GE3: Cost to maintain	2	Cost to maintain is low.
Physical (EPI)	EP1: Visual distraction	3	Visual distraction is medium.
	EP2: Auditory distraction	3	Auditory distraction is medium.
Social (ESI)	ES1: Interaction	4	Social interaction is high.
Role (SRI)	SR1: Role type	4	Role type is facilitator or group.
	SR2: Role priority	5	Role priority is high.

The application of context-based framework to all tasks of PBL are shown in detail in Tables A1- A12 in Appendix A.

After applying context-based framework to all PBL tasks (T1-T12), the weighted mean values for characteristics of the system are calculated. In Table 14, the summary of all characteristics of context-based framework is shown. Actual values are calculated using Equation 1.

Table 14. Summary of the context-based framework expected characteristics vs. actual values

Indicator of Requirement Factor	Actual Value	Expected Value	Comment
STT1	3.44	3	T1, T9, T10 and T11 are scheduled.
STT2	3.80	3	T2, T3, T4, T9, T10 and T11 are event-triggered.
STT3	4.60	5	T1 and T6 have one-way information flow.
STT4	2.66	2	T1, T2 and T10 need at least audio.
STT5	2.72	3	T5 and T6 have long task lifetime.
STL1	2.92	3	T10 and T11 are location dependent.
STL2	2.76	3	T5, T10 and T11 need location report.
STL3	1.32	3	T5 needs available services.
PE1	3.36	2	T1, T2 need high user experience.

TS1	3.84	3	All tasks are at least partially decomposable.
TS2	4.04	3	T1 and T6 are not sequential.
TS3	3.92	3	T1 and T6 are not interdependent.
TS4	4.16	5	T6 has no collaboration.
TB1	3.04	3	T1, T4, T10, T11 and T12 are not repeated.
TB2	2.52	3	T5, T6, T7, T8, T9 and T12 do not need notification.
TC1	4.28	3	Most tasks have high complexity.
GSec1	2.92	2	T5 has low identification/authentication need. T6 has none.
GSec2	2.92	2	T5 has low authorization/access control need. T6 has none.
GSec3	1.36	2	T1 and T5 have medium data integrity need.
GSec4	1.00	1	None of the tasks need confidential data.
GSec5	2.56	2	T1 has high non-repudiation.
GSca1	1.92	3	T5 has high number of nodes.
GSca2	4.08	3	T5 has high density.
GSca3	4.08	3	T5 has high traffic.
GT1	3.52	4	All tasks should have at least medium data transparency.
GT2	3.36	4	All tasks should have at least medium communication transparency.
GA1	3.42	4	T1, T5, T6 and T10 cannot be executed if computer is down.
GA2	1.40	3	T1 and T10 have medium network downtime tolerance.
GA3	1.40	3	T1 and T10 have medium network not available tolerance.
GE1	2.02	2	T6 has medium running cost.
GE2	3.00	3	All tasks have medium setup cost.
GE3	1.98	2	T5 has high maintenance cost.
EP1	3.00	3	All tasks have medium visual distraction.
EP2	3.00	3	All tasks have medium auditory distraction.
ES1	4.14	4	T6 has low social interaction.
SR1	4.44	4	T5 and T6 are individual tasks.

SR2	5.00	5	All tasks are executed by primary roles.
-----	------	---	--

Table 15 shows the actual and expected values for requirement indicators for attributes of the system. Actual values are calculated using Equation 2.

Table 15. Summary of the expected indicators of requirement attributes vs. actual values

Indicator of Requirement Attribute	Actual Value	Expected Value
Direct Indicators		
Spatio-temporal Time Indicator (STI)	2.87	3
Spatio-temporal Location Indicator (SLI)	2.33	3
Personal Expertise Indicator (PEI)	2	2
Task Structure Indicator (TSI)	2.82	3
Task Behavior Indicator (TBI)	2.76	3
Task Complexity Indicator (TCI)	3	3
General Security Indicator (GSI)	1.85	2
General Scalability Indicator (GScal)	3.52	4
General Transparency Indicator (GTI)	3.44	4
General Availability Indicator (GAI)	2.48	4
General Efficiency Indicator (GEI)	2.99	3
Environmental Physical Indicator (EPI)	3	3
Environmental Social Indicator (ESI)	3	4
Social Role Indicator (SRI)	4	4
Derived Indicators		
Network Capacity Indicator (NCI)	3.43	4
Mobility Indicator (MI)	3.98	4
Transactional Support Indicator (TSI)	3.41	3
Energy Consumption Indicator (ECI)	2.32	3

Table 16. Summary of the system target profile

Indicator of Requirement Factor (Context)	Expected value for indicator of requirement factor	Meets expected value	Indicator of Requirement Attribute	Meets expected value
Spatio-temporal	Satisfied	No	Time (STI)	Almost
			Location (SLI)	
Personal	Satisfied	Yes	Expertise (PEI)	
Task	Satisfied	No	Structure (TSI)	Almost
			Behavior (TBI)	
			Complexity (TCI)	
General	Satisfied	No	Security (GSI)	Almost
			Scalability (GScal)	
			Transparency (GTI)	
			Availability (GAI)	
			Efficiency (GEI)	Almost
Environmental	Satisfied	No	Physical (EPI)	
			Social (ESI)	
Social	Satisfied	Yes	Role (SRI)	
	Satisfied	No	Network Capacity (NCI)	
	Satisfied	No	Mobility (MI)	Almost
	Satisfied	No	Transactional Support (TSI)	
	Satisfied	No	Energy Consumption (ECI)	

Table 16 summarizes the results for system level indicators of system attributes and factors for PBL. The values are calculated using the rules from Table 6-7. From this table, it is easy to see which requirements are met and which ones need to be changed/improved. If a requirement indicator does not meet its expected value, it will

affect the related context requirement as well. In order to change/improve requirement indicator, the characteristics that are related to the requirement should be investigated.

Since spatio-temporal time (STI) and location (SLI) requirement indicators did not meet their expected values, the context requirement is affected. To trace the problem back to the characteristic level for the time indicator (STI), it is easily seen that STT3 is the cause. STT3 for system level is expected to be 5 (the two-way information flow). However, the actual value is computed to be 4.60. This value indicates the information flow type to be between one-way and two-way. STT3 has to be changed for tasks T1 and T6 to reach the expected value for STI.

Spatio-temporal location indicator's (SLI) expected value is 3, indicating that system is partially location-dependent and it requires services for some tasks. This value is not met because of characteristics STL1, STL2 and mostly STL3. Since some of the tasks do not require service availability, STL3 did not meet its expected value of 3.

Personal expertise indicator (PEI) does not need to be changed/improved since it has met its expected value.

For task context, small changes need to be done to task structure indicator (TSI) and task behavior indicator (TBI). Their actual values are very close to their expected values. For TSI, collaboration type characteristic (TS4) requires small improvements. For TBI, task completion/failure characteristic (TB2) needs to be improved. Task complexity indicator (TCI) does not require any changes to its characteristics because the expected value is met.

General context requirements are found out to be not satisfactory. Although general security (GSI) and efficiency (GEI) indicators' actual values are very close to their expected values, big changes for scalability, transparency, and availability requirements has to be implemented. With the current values, system does not reach its scalability, transparency and availability requirements.

Environmental physical (EPI), social role (SRI), and mobility (MI) indicators all satisfy their expected values. Small improvements can be implemented to SRI and MI. Network capacity indicator (NCI) needs changes as well as environmental social indicator (ESI).

Transactional support indicator (TSI) does not reach its expected value. This indicates that PBL mobile system should be able to support more relaxed ACID properties. STT1-STT3 characteristics of the mobile system have actual values greater than the expected values.

Energy consumption indicator (ECI) falls short of its expected value as well. STL3 characteristic has the most effect on this result. STL3 for the PBL mobile system has to be examined and improved.

Figure 7 and 8 show the comparison of actual values of requirement indicators with their expected values. Figures 9-16 show the comparison of each characteristic with their expected values.

Figure 7. Comparison of requirement indicators (radar chart)

Figure 8. Comparison of requirement indicators (line-column chart)

Figure 9. Comparison of characteristics

Figure 10. Comparison of characteristics (cont.)

Figure 11. Comparison of characteristics (cont.)

Figure 12. Comparison of characteristics (cont.)

Figure 13. Comparison of characteristics (cont.)

Figure 14. Comparison of characteristics (cont.)

Figure 15. Comparison of characteristics (cont.)

Figure 16. Comparison of characteristics (cont.)

5.3 MOWAHS Framework Applied to PBL

In this section, MOWAHS framework is applied to PBL in order to compare the results with context-based framework. Table 17 shows the summary of MOWAHS framework applied to PBL. Since the MOWAHS framework is used for comparison purposes, the details of the process steps are not shown.

Table 17. MOWAHS framework applied to PBL tasks

Char.	T1	T2	T3	T4	T5	T6
Weight	5 Very high	4 High	3 Medium	3 Medium	4 High	5 Very high
G1	1 No	1 No	5 Yes	5 Yes	5 Yes	5 Yes
G2	1 No	3 Partial	5 Yes	5 Yes	5 Yes	1 No
G3	1 Planned	5 Ad-hoc	5 Ad-hoc	5 Ad-hoc	5 Ad-hoc	5 Ad-hoc
G4	3 After task	3 After task	5 Within task	5 Within task	1 No	1 No
G5	3 Once	3 Once	3 Once	3 Once	5 Many	5 Many
I1	5 Video	4 Audio	2 Text	2 Text	2 Text	2 Text
I2	5 Continuous	5 Continuous	5 Continuous	5 Continuous	3 Discrete	3 Discrete

I3	5 Real-time	5 Real-time	4 High	4 High	3 Medium	4 High
I4	5 Real-time	5 Real-time	4 High	4 High	3 Medium	4 High
I5	5 Very fast	5 Very fast	4 Fast	4 Fast	3 Medium	3 Medium
L1	3 Partial	3 Partial	3 Partial	1 No	1 No	1 No
L2	1 No	1 No	1 No	1 No	5 Yes	1 No
L3	1 No	1 No	1 No	1 No	1 No	1 No
L4	1 No	1 No	1 No	1 No	5 Yes	1 No
L5	5 Yes	5 Yes	5 Yes	5 Yes	3 Partial	3 Partial
T1	1 No	5 Yes	5 Yes	5 Yes	3 Partial	3 Partial
T2	5 Yes	3 Partial	3 Partial	3 Partial	1 No	1 No
T3	1 No	1 No	3 Partial	5 Yes	1 No	1 No
T4	1 No	1 No	3 Partial	1 No	5 Yes	5 Yes
T5	2 Minutes	2 Minutes	3 Hours	2 Minutes	4 Days	4 Days

Char.	T7	T8	T9	T10	T11	T12
Weight	4 High	4 High	5 Very high	5 Very high	5 Very high	3 Medium
G1	3 Partial	3 Partial	5 Yes	5 Yes	5 Yes	3 Partial
G2	5 Yes	5 Yes	5 Yes	5 Yes	5 Yes	5 Yes
G3	5 Ad-hoc	5 Ad-hoc	5 Ad-hoc	1 Planned	5 Ad-hoc	5 Ad-hoc
G4	1 No	1 No	3 After task	5 Within task	5 Within task	5 Within task
G5	5 Many	3 Once	3 Once	3 Once	3 Once	3 Once
I1	2 Text	2 Text	2 Text	4 Audio	2 Text	2 Text
I2	5 Continuous	5 Continuous	5 Continuous	5 Continuous	5 Continuous	5 Continuous
I3	4 High	4 High	4 high	5 Real-time	5 Real-time	4 High
I4	4 High	4 High	4 High	5 Real-time	5 Real-time	4 High
I5	3 Medium	4 High	4 High	5 Very fast	5 Very fast	5 Very fast
L1	3 Partial	3 Partial	3 Partial	5 Yes	5 Yes	3 Partial
L2	1 No	1 No	1 No	1 No	1 No	1 No
L3	1 No	1 No	1 No	5 Yes	1 No	1 No
L4	3 Partial	3 Partial	3 Partial	5 Yes	5 Yes	3 Partial
L5	3 Partial	3 Partial	5 Yes	5 Yes	5 Yes	5 Yes
T1	3 Partial	3 Partial	5 Yes	5 Yes	5 Yes	3 Partial
T2	3 Partial	3 Partial	5 Yes	5 Yes	5 Yes	3 Partial

T3	1 No	3 Partial	3 Partial	5 Yes	5 Yes	3 Partial
T4	5 Yes	5 Yes	5 Yes	1 No	1 No	5 Yes
T5	4 Days	3 Hours	3 Hours	2 Minutes	3 Hours	2 Minutes

After applying MOWAHS to PBL, the average values are calculated for all tasks

(T1-T12). Table 18 shows calculated results for the whole system.

Table 18. Summary of the MOWAHS framework characteristics

Characteristics	Result	Comment
G1	3.83	T1 and T2 are not decomposable.
G2	4.16	T1 and T6 are not sequential.
G3	4.33	T1 and T10 are planned.
G4	3.16	T3, T4, T10, T11 and T12 have data synchronization within task.
G5	3.50	T5, T6 and T7 have many data exchanges during their lifetimes.
I1	2.58	T1, T2 and T10 need at least audio.
I2	4.66	T5 and T6 need discrete information streaming. All other tasks have continuous streaming.
I3	4.25	T1, T2, T10 and T11 require real-time data.
I4	4.25	T1, T2, T10 and T11 produce real-time data.
I5	4.16	T1, T2, T10, T11 and T12 need very fast transmission speed.
L1	2.83	T10 and T11 are location dependent.
L2	1.33	T5 needs available services.
L3	1.33	T10 produces services.
L4	2.66	T5, T10 and T11 need location report.
L5	4.33	All tasks need at least partial route constraints.
T1	3.83	T2, T3, T4, T9, T10 and T11 are event-triggered.
T2	3.33	T1, T9, T10 and T11 are scheduled.
T3	2.66	T4, T10 and T11 need coordination with other tasks.
T4	3.16	T5, T6, T7, T8, T9 and T11 have task resumption.
T5	2.83	T5, T6 and T7 have task lifetime of days.

After the calculation of system characteristics, requirement indicators are calculated. The main difference between the MOWAHS and context-based frameworks is that MOWAHS framework does not take weight of the task into consideration when

calculating requirement indicators. Summary of requirement indicators is shown in Table 19.

Table 19. Summary of MOWAHS requirement indicators

Indicator	Value
General Task Indicator (GTI)	3.8
Information Complexity Indicator (ICI)	3.98
Location Complexity Indicator (LCI)	2.56
Time Complexity Indicator (TCI)	3.16
Network Connectivity Indicator (NCI)	3.50
Network Speed Indicator (NSI)	4.22
Energy Consumption Indicator (ECI)	2.16
Transactional Support Indicator (TSI)	3.47
Mobility Indicator (MI)	3.50

6.0 FINDINGS AND RESULTS

This section of the thesis focuses on the findings and comparisons of both context-based and MOWAHS frameworks.

As indicated earlier, context-based framework is based on MOWAHS. However, there are many changes applied to the proposed framework. The main difference between the two frameworks is that MOWAHS uses tasks as the start point of the requirements engineering process whereas context-based framework takes different contexts into consideration. Context-based framework starts the RE process with user tasks with respect to the six different contexts. Since mobile systems are desired to be context-aware, using context-related characteristics provide more information about the system.

Context-based framework provides accurate and more detailed information about system requirements and problem areas to be changed/improved. Context-based framework uses the system target profile with expected values to provide a more scientific way to approach the requirements. When requirement indicators are calculated, they are compared with the expected values. Using this method provides more information on the proximity of actual requirement values. Also, it is easier and more efficient to trace problems back to characteristics that cause the inaccurate values for the requirement. However, MOWAHS framework computes average values for system characteristics. Task weights are not taken into consideration when computing the requirement indicators. MOWAHS framework assumes that each characteristic affects

the system requirement indicators the same way. Therefore requirement indicators are computed by taking the average of related characteristics. However, context-based framework assumes that not every characteristic is the same for every mobile system. Figure 17 shows the summary of both frameworks' requirement indicators. It is clear to see that context-based framework provides more detailed information on system requirements. Task structure and task complexity indicators that are calculated using the MOWAHS framework, miss their target. This is because of MOWAHS way of calculating the requirement indicators by taking the average of indicator characteristics. Context-based framework reaches a more desirable value by taking weights into consideration. It is also clear that context-based framework covers more requirement indicators than the MOWAHS framework. MOWAHS falls short of providing information about security, scalability, availability, efficiency, environmental, and social requirements of the mobile system.

Figure 17. Comparison of context-based and MOWAHS frameworks

7.0 CONCLUSION AND RECOMMENDATIONS

Mobile systems have become more common in everyday life because of the need for being mobile and still being able to reach and use information. As indicated throughout this thesis, there is no specific approach to the requirements engineering (RE) process for mobile systems. This makes it very time consuming and almost impossible to build a quality mobile system. Seeing the problems in this area, this study tries to establish a requirements engineering process specific to mobile systems.

As a result of this study, a requirements engineering process specific to mobile systems is developed. A context-based framework is proposed to characterize the different contexts of the mobile system. From this characterization, requirements are elicited, analyzed, validated, changed, and improved using a more mathematical model.

Proposed context-based framework is based on another framework called MOWAHS framework. Context-based framework takes MOWAHS framework and improves the characteristics belonging to different user contexts. Both frameworks are explained in detail throughout this document.

In this study, more characteristics are added to context-based framework. Although time and location attributes of spatio-temporal context are based on MOWAHS framework, some characteristics are changed due to our belief that these characteristics are related more to context-aware computing. Personal, task, environmental, and social contexts and their attributes are all taken from context-aware computing concept. Their characteristics are created using the information from related papers. Security,

scalability, transparency, availability, and efficiency attributes are created in order to fill void for related system requirements. MOWAHS framework falls short for establishing requirements for these attributes that are a very important part of mobile systems.

Both frameworks are applied to a problem-based learning (PBL) mobile system in order to see if they succeed in gathering functional and non-functional requirements of the system. It is shown that both frameworks can be applied to a mobile system successfully to derive system requirements. However there are some important differences between these frameworks. Context-based framework points to problem areas and uses a mathematical model that provides a specific way to trace back to problem areas where requirements need to be changed/improved. RE process steps are repeated until expected requirement values of the system target profile are met. On the other hand, MOWAHS framework establishes some requirement indicators but there are no expected values for comparison. MOWAHS framework also assumes that each characteristic affects the mobile system in the same way. Context-based framework recognizes the important characteristics for different mobile systems and takes this into consideration. Context-based framework is also scalable. Not all the characteristics are applicable to all systems. Flexibility of the context-based framework allows the addition of new contexts and characteristics as well as removal of the unnecessary ones.

Although context-based framework is successfully applied to a case study, it is not used for a real life project to prove its validity. The following are recommended for further studies to validate this framework:

- Using context-based framework for more real life mobile systems during the requirements engineering step of the system development life cycle.
- Applying context-based framework to hypothetical mobile systems in order to improve the framework.
- Applying context-based framework to already implemented mobile systems to compare the derived requirements with the implemented requirements.
- Evaluating the strengths and weaknesses of the context-based framework by collecting information from the suggested implementations above.
- Adding different characteristics and contexts to the framework.

References

- Agre, P. E. (2001). Changing Places: Contexts of Awareness in Computing [Electronic version]. *Human-Computer Interaction*, 16 (2-4), 177-192.
- Armarego, J. (2004). Student perceptions of quality learning: evaluating PBL in Software Engineering. Paper presented at the Seeking Educational Excellence: 13th Teaching Learning Forum. Retrieved May 4, 2004, from http://eng.murdoch.edu.au/~jocelyn/papers/armaregojTLF2004_final.pdf
- Barrows, H. S. (1988). *The Tutorial Process*. Springfield, IL: Southern Illinois University School of Medicine.
- Barrows, H. S., Tamblyn, R.M. (1980) *Problem-Based Learning: An Approach to Medical Education*. New York: Springer Publishing Company.
- Basole, R. (2004). The Value and Impact of Mobile Information and Communication Technologies. Paper presented at the 2004 IFAC Symposium. Retrieved March 7, 2005, from <http://www.ti.gatech.edu/docs/BasoleIFAC2004MobileEnterprises.pdf>
- Brackett, J. W. (1990) *Software Requirements, Technical Report SEI-CM-19-1.2*. Retrieved October 18, 2004, from <http://www.sei.cmu.edu/pub/documents/90.reports/pdf/cm19.pdf>
- Bradley, N. A., Dunlop, M. D. (2002). Understanding Contextual Interactions to Design Navigational Context-Aware Applications. Paper presented at the 2002 Proceedings of Mobile HCI. Retrieved June 7, 2004, from http://www.gloss.cs.strath.ac.uk/dissemination/Bradley_MobileHCI03.pdf
- Brooks, F. P. (1987). No silver bullet: Essence and Accidents of Software Engineering [Electronic version]. *IEEE Computer*, 20 (4), 10-19.
- Burrell, J., Treadwell, P., Gay, G. K. (2000). Designing for Context: Usability in a Ubiquitous Environment. Paper presented at the Proceedings on the 2000 conference on Universal Usability. Retrieved June 7, 2004, from http://www.hci.cornell.edu/LabArticles/DesigningContext_Burrell.pdf
- Carroll, J., Kjeldskov, J., Tobin, D., Vetere, F. (2003). A User-centered Process for Determining Requirements for Mobile Technologies: the TramMate Project. Paper presented at the Seventh Pacific Asia Conference on Information Systems. Retrieved February 16, 2004, from <http://www.pacis2003.org/proceedings/papers/business-Processes/136.PDF>

- Chalmers, D., Sloman, M. (1999). QoS and Context Awareness for Mobile Computing. Paper presented at the First International Symposium on Handheld and Ubiquitous Computing. Retrieved October 31, 2003, from http://www.informatics.sussex.ac.uk/users/dc52/Papers/QoSAndContext_FieldworkWorkshop99.pdf
- Churchill, E., Munro, A. (2001). Work/Place: Mobile Technologies and Arenas of Activity [Electronic version]. *ACM SIGGROUP Bulletin December 2001, Volume 22 Issue 3*.
- Davies, N., Blair, G., Cheverst, K., Friday, A. (1996). Supporting Collaborative Applications in a Heterogeneous Mobile Environment [Electronic version]. *Computer Communications Special Issue on Mobile Computing*, 19, 346-358.
- De Florio, V., Deconinck, G. (2002). On Some Key Requirements of Mobile Application Software. Paper presented at the Ninth Annual IEEE International Conference and Workshop on the Engineering of Computer-Based Systems. Retrieved October 10, 2003, from <http://dlib.computer.org/conferen/ecbs/1549/pdf/15490003.pdf>
- Delaney, J. D., Mitchell, G. G. (2002). PBL applied to Software Engineering Group Projects. Paper presented at the International Conference on Information and Communication in Education. Retrieved May 4, 2004, from www.cs.may.ie/~georgem/papers/PBL2002.ps
- Dey, A. (2001). Understanding and Using Context [Electronic version]. *Personal and Ubiquitous Computing Journal*, (1), 4-7.
- Dey, A., Abowd, G. (1999). Towards a Better Understanding of Context and Context-Awareness. Paper presented at the 2000 Conference on Human Factors in Computing Systems. Retrieved April 26, 2004, from <ftp://ftp.cc.gatech.edu/pub/gvu/tr/1999/99-22.pdf>
- Finkelstein, A., Savigni, A. (2001). A Framework for Requirements Engineering for Context-Aware Services. Paper presented at First International Workshop From Software Requirements to Architectures (STRAW 01) 23rd International Conference on Software Engineering. Retrieved February 16, 2004, from <http://www.cs.ucl.ac.uk/staff/A.Finkelstein/papers/straw.pdf>
- Galal, G. H., J. T. McDonnell and R. J. Paul (2000). The Role of Interpretive Evaluation in Engineering Information Systems Requirements. Paper presented at the 2000 American Conference on Information Systems. Retrieved October 28, 2003, from <http://is.lse.ac.uk/Support/AMCIS/AMCIS2000/pdf/files/papers/215.pdf>

- Goker, A., Myrhaug, H. I. (2002). User context and personalization. Paper presented at the 2002 ECCBR Workshop on Case Based Reasoning and Personalization. Retrieved February 16, 2004, from <http://www.smartweb.rgu.ac.uk/papers/AGoker.pdf>
- Hall, E. (1962). *The Hidden Dimension*, New York, NY: Anchor Press.
- Hall, E. (1959). *The Silent Language*, New York, NY: Double-Day.
- Kakihara, M., Sorensen, C. (2002). Mobility: An Extended Perspective. Paper presented at the Thirty-Fifth Hawaii International Conference on System Sciences (HICSS-35). Retrieved March 4, 2005, from <http://mobility.is.lse.ac.uk/download/Kakihara&Sorensen2002.pdf>
- Kofod-Petersen, A., Aamodt, A. (2003). Case-Based Situation Assessment in a Mobile Context-Aware System. Paper presented at the 2003 Artificial Intelligence in Mobile System Conference. Retrieved March 15, 2004, from <http://w5.cs.uni-sb.de/~krueger/aims2003/camera-ready/kofod-petersen-6.pdf>
- Krogstie, J. (2001). Requirement Engineering for Mobile Information Systems. Paper presented at the Seventh International Workshop on Requirements Engineering: Foundation for Software Quality. Retrieved October 26, 2003, from <http://www.ifi.uib.no/conf/refsq2001/papers/p6.pdf>
- Lauesen, S. (2003). Task Descriptions as Functional Requirements [Electronic version]. *IEEE Software*, 20(2), 58-65.
- Maccari, A. (1999). The Challenges of Requirements Engineering in Mobile Telephones Industry. Paper presented at the Tenth International Workshop on Database and Expert Systems Applications. Retrieved October 27, 2003, from <http://dlib.computer.org/conferen/dexa/0281/pdf/02810336.pdf>
- McPhee, C., Eberlein, A. (2002). Requirements Engineering for Time-to-Market Projects. Paper presented at the Ninth Annual IEEE International Conference and Workshop on the Engineering of Computer-Based Systems. Retrieved September 17, 2003, from <http://csdl.computer.org/dl/proceedings/ecbs/2002/1549/00/15490017.pdf>
- Moran, T. P., Dourish P. (2001) Context Aware Computing [Electronic version]. *Human-Computer Interaction*, 6, 3-8.
- Munoz, M., Rodriguez, M., Favela, J., Gonzalez, V.M., Martinez-Garcia, A.I. (2003). Context-aware mobile communication in hospitals [Electronic version]. *IEEE Computer*, 36 (8), 60-67.

- Ogata, H., Yano, Y. (2004). Context-Aware Support for Computer Supported Ubiquitous Learning. Paper presented at the Second IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE). Retrieved June 1, 2004, from <http://www-yano.is.tokushima-u.ac.jp/ogata/clue/WMTE-03-1-50.pdf>
- Papavassiliou, S., Xu, S., Orlik, P., Snyder, M., Sass, P. (2002). Scalability in Global Mobile Information Systems (GloMo): Issues, Evaluation Methodology and Experiences [Electronic version]. *Wireless Networks*, 8(6), 637-648.
- Patterson, C. A., Muntz, R.R., Pancake C. M. (2003). Challenges in Location-Aware Computing [Electronic version]. *IEEE Pervasive Computing*, 2 (2), 80-89.
- Pica, D., C. Sorensen, & D. Allen (2004). On Mobility and Context of Work: Exploring Mobile Police Work. Paper presented at Thirty-Seventh Hawaii International Conference on System Sciences (HICSS-37). Retrieved March 4, 2005, from <http://is.lse.ac.uk/staff/sorensen/download/PicaSorensenAllen.pdf>
- Ramampiaro, H., Wang, A. I., Sorensen, C. F., Nam Le, H., Nygard, M., Conradi, R. (2003). Requirement Indicators for Mobile Work: The MOWAHS Approach. Paper presented at IASTED International Conference on Applied Informatics. Retrieved February 16, 2003, from <http://www.idi.ntnu.no/grupper/su/publ/carlfrs/ai2003-mowahs-indicator.pdf>
- searchMobileComputing web page (2003). Retrieved October 9, 2004, from http://searchmobilecomputing.techtarget.com/sDefinition/0,,sid40_gci213462,00.html
- Shen, X., Shen, J. (2001). User Requirements in Mobile Systems. Paper presented at the Seventh Americas Conference on Information Systems. Retrieved October 27, 2003, from <http://web.njit.edu/~jxs1866/papers/amcis01-requirements.PDF>
- Sorensen, C. F., Wang, A. I., Nam Le, H., Ramampiaro, H., Nygard, M., Conradi, R. (2002). The MOWAHS Characterization Framework for Mobile Work. Paper presented at IASTED International Conference on Applied Informatics. Retrieved February 16, 2003, from <http://www.idi.ntnu.no/grupper/su/publ/pdf/ai2002-mowahs-framework.pdf>
- Sutcliffe, A. (2003). Symbiosis and synergy? scenarios, task analysis and reuse of HCI knowledge [Electronic version]. *Interacting with Computers, Volume 15, Issue 2*, 245-26.
- Sutcliffe, A. (1997). Task-related Information Analysis [Electronic version]. *International Journal of Human-Computer Studies, Volume 47, Issue 2*, 223-257.

- Tsalgaidou, A., Pitoura, E. (2001). Business Models and Transactions in Mobile Electronic Commerce: Requirements and Properties [Electronic version]. *Journal of Computer Networks, Special Issue on Electronic Business Systems*, 37 (2), 221-236.
- Tsalgaidou, A., Veijalainen, J., Markkula, J., Katasonov, A., Hadjiefthymiades, S. (2003). Mobile E-Commerce and Location-Based Services: Technology and Requirements. Paper presented at the 9th Scandinavian Research Conference on Geographical Information Science. Retrieved November 3, 2003, from <http://www.scangis.org/scangis2003/papers/27.pdf>
- Uden, L., Dix, A. (2000). Lifelong Learning for Software Engineers. Paper presented at IFIP World Computer Congress - ICEUT2000 Educational Uses of Communication and Information Technologies. Retrieved June 1, 2004, from <http://www.comp.lancs.ac.uk/computing/users/dixa/papers/wcc-pbl-2000/wcc-pbl-2000.pdf>
- UsabilityNet web page (2005). Retrieved March 7, 2005, from <http://www.usabilitynet.org/tools/taskanalysis.htm>
- Wikipedia, the free encyclopedia web page (2004). Retrieved November 1, 2004 from <http://en.wikipedia.org/wiki/ACID>

Appendix A. Context-based Framework Application Tables for PBL Tasks

Task: T1 – Present problem

Table A1. Context-based framework applied to T1

Characteristic	T1	Details
Weight	5 Very high	Problem presentation is the most important task. Without a clear problem statement, it is very difficult to find the appropriate solution.
STT1	5 Yes	T1 is a scheduled task.
STT2	1 No	T1 is not event-triggered.
STT3	3 One-way	Information flow is from the facilitator to students.
STT4	5 Video	Information flow needs to be video.
STT5	1 Short	T1 task lifetime is short compared to other tasks.
STL1	3 Partial	Location dependency is partial because as long as the students learn about the problem real-time, they do not have to be present in a specific location.
STL2	1 No	Location report is not needed during T1.
STL3	1 No	No services are needed during T1.
PE1	5 High	User expertise should be high. Users should know and be familiar about the PBL.
TS1	1 No	T1 is not decomposable. It is one task.
TS2	1 No	T1 is not sequential. It does not follow another task.
TS3	1 No	T1 is not dependant on other tasks.
TS4	4 Broadcast	Problem presentation should be broadcasted to all students.
TB1	1 No	T1 is not repeated during PBL.
TB2	5 Yes	Task completion/failure notification is needed since each student should successfully learn about the problem.
TC1	3 Medium	T1 has medium complexity since it is a one-way communication.
GSec1	5 High	System needs to make sure that the true facilitator presents the problem.
GSec2	5 High	System also needs to make sure that authorized students can access the system.
GSec3	3 Medium	System should provide data integrity but it is not need to be high since it is just the problem presentation.
GSec4	1 N/A	Confidentiality is not needed.

GSec5	5 High	Non-repudiation is needed since facilitator needs to know all the students received the problem statement.
GSca1	3 Medium	Number of nodes is medium.
GSca2	3 Medium	Density is medium.
GSca3	3 Medium	Traffic is medium.
GT1	5 High	System should support transparent data between the task executors. Data should be portable.
GT2	5 High	System should support transparent communication between different devices.
GA1	1 Low	When computer is down, task cannot be executed.
GA2	3 Medium	When network is down, task can be executed but it may not be completed.
GA3	3 Medium	When network is not available, task can be executed but it may not be completed.
GE1	2 Low	Cost is low to run T1.
GE2	3 Medium	Cost to setup T1 is medium since Audio or Video may be involved. These require high network capacity.
GE3	2 Low	Cost of maintaining the system for T1 is low.
EP1	3 Medium	Visual distraction during T1 is medium.
EP2	3 Medium	Auditory distraction during T1 is medium.
ES1	3 Medium	Social interaction during T1 is medium.
SR1	4 Facilitator	The facilitator executes T1.
SR2	5 Primary	Facilitator has the primary role.

Task: T2 – Set hypothesis**Table A2. Context-based framework applied to T2**

Characteristic	T2	Details
Weight	4 High	Setting hypothesis has high importance.
STT1	3 Partial	T2 is a partially scheduled task. It should occur after T1.
STT2	5 Yes	T2 is event-triggered (triggered by T1).
STT3	5 Two-way	Information flow is two-way between the facilitator and the students.
STT4	4 Audio	Information flow needs to be audio at least.
STT5	3 Medium	T2 task lifetime is medium compared to other tasks.
STL1	3 Partial	Location dependency is partial because as long as the students set the hypothesis, they do not have to be present in a specific location.
STL2	1 No	Location report is not needed during T2.
STL3	1 No	No services are needed during T2.
PE1	5 High	User expertise should be high. Users should know and be familiar about the PBL in order to carry the PBL process.
TS1	1 No	T2 is not decomposable. It is one task.
TS2	3 Partial	T2 is partially sequential. It follows T1.
TS3	5 Yes	T2 is dependant on other tasks.
TS4	5 Ad-hoc	Communication should be available between the group members.
TB1	3 Partial	T2 can be repeated during PBL.
TB2	3 Partial	Task completion/failure notification is partially needed but it is not essential.
TC1	5 High	T2 has high complexity since it is a two-way communication between interested parties.
GSec1	3 Medium	System needs to make sure that the students from the same group communicate.
GSec2	3 Medium	System also needs to make sure that authorized students of same group can access the system.
GSec3	1 N/A	Data integrity is not needed in T2.
GSec4	1 N/A	Confidentiality is not needed in T2.
GSec5	1 N/A	Non-repudiation is not needed in T2.
GScal	1 Low	Number of nodes is low (group members).
GScal2	5 Low	Density is low.
GScal3	5 Low	Traffic is low.

GT1	5 High	System should support transparent data between the task executers. Data should be portable.
GT2	5 High	System should support transparent communication between different devices.
GA1	5 High	When one of the computers in the network is down, T2 can still be executed.
GA2	1 Low	When network is down, T2 cannot be executed. T2 requires a two-way, ad-hoc communication. Network has to be up.
GA3	1 Low	T2 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T2 is medium.
EP2	3 Medium	Auditory distraction during T2 is medium.
ES1	5 High	Social interaction during T2 is high between group members.
SR1	5 Group	The group members initiate T2.
SR2	5 Primary	Group members have the primary role.

Task: T3 – Identify learning

Table A3. Context-based framework applied to T3

Characteristic	T3	Details
Weight	3 Medium	Identifying the learning has medium importance.
STT1	3 Partial	T3 is a partially scheduled task. It should occur after T2.
STT2	5 Yes	T3 is event-triggered (triggered by T2).
STT3	5 Two-way	Information flow is two-way between the facilitator and the students.
STT4	2 Text	Information flow needs to be text at least.
STT5	3 Medium	T3 task lifetime is medium compared to other tasks.
STL1	3 Partial	Location dependency is partial because as long as the students communicate with each other and the facilitator, they do not have to be present in a specific location.
STL2	1 No	Location report is not needed during T3.
STL3	1 No	Services are not needed during T3.
PE1	3 Medium	User expertise should be medium.
TS1	5 Yes	T3 is decomposable. It has sub tasks.
TS2	5 Yes	T3 is sequential. It follows T2.
TS3	5 Yes	T3 is dependant on other tasks.
TS4	5 Ad-hoc	Communication should be available between the group members.
TB1	3 Partial	T3 can be repeated during PBL.
TB2	3 Partial	Task completion/failure notification is partially needed but it is not essential.
TC1	5 High	T3 has high complexity since it is a two-way communication between interested parties.
GSec1	3 Medium	System needs to make sure that the students from the same group communicate.
GSec2	3 Medium	System also needs to make sure that authorized students of same group can access the system.
GSec3	1 N/A	Data integrity is not needed in T3.
GSec4	1 N/A	Confidentiality is not needed in T3.
GSec5	1 N/A	Non-repudiation is not needed in T3.
GScal	1 Low	Number of nodes is low (group members).
GScal2	5 Low	Density is low.
GScal3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executors. Data should be portable.

GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	5 High	When one of the computers in the network is down, T3 can still be executed.
GA2	1 Low	When network is down, T3 cannot be executed. Since T3 requires a two-way, ad-hoc communication, network has to be up.
GA3	1 Low	T3 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T3 is medium.
EP2	3 Medium	Auditory distraction during T3 is medium.
ES1	5 High	Social interaction during T3 is high between group members.
SR1	5 Group	The group members initiate T3.
SR2	5 Primary	Group members have the primary role.

Task: T4 - Assign tasks**Table A4. Context-based framework applied to T4**

Characteristic	T4	Details
Weight	3 Medium	Assigning tasks has medium importance.
STT1	3 Partial	T4 is a partially scheduled task. It should occur after T3.
STT2	5 Yes	T4 is event-triggered (triggered by T3).
STT3	5 Two-way	Information flow is two-way between the facilitator and the students.
STT4	2 Text	Information flow needs to be text at least.
STT5	1 Short	T4 task lifetime is short compared to other tasks.
STL1	1 No	Location dependency is not needed.
STL2	1 No	Location report is not needed during T4.
STL3	1 No	Services are not needed during T4.
PE1	3 Medium	User expertise should be medium.
TS1	5 Yes	T4 is decomposable. It has sub tasks (task list, prioritize tasks, assign tasks).
TS2	5 Yes	T4 is sequential. It follows T3.
TS3	3 Partial	T4 is partially dependant on other tasks.
TS4	5 Ad-hoc	Communication should be available between the group members.
TB1	1 No	T4 is not repeated during PBL.
TB2	5 Yes	Task completion/failure notification is needed since T4 is essential for the rest of the tasks. Assignment should be clear.
TC1	5 High	T4 has high complexity since it is a two-way communication between interested parties.
GSec1	3 Medium	System needs to make sure that the students from the same group communicate.
GSec2	3 Medium	System also needs to make sure that authorized students of same group can access the system.
GSec3	1 N/A	Data integrity is not needed in T4.
GSec4	1 N/A	Confidentiality is not needed in T4.
GSec5	3 Medium	Non-repudiation is needed in T4. Each group member should be aware of assigned tasks.
GScal	1 Low	Number of nodes is low (group members).
GScal2	5 Low	Density is low.
GScal3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.

GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	5 High	When one of the computers in the network is down, T4 can still be executed.
GA2	1 Low	When network is down, T4 cannot be executed. T4 requires a two-way, ad-hoc communication.
GA3	1 Low	T4 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T4 is medium.
EP2	3 Medium	Auditory distraction during T4 is medium.
ES1	5 High	Social interaction during T4 is high between group members.
SR1	5 Group	The group members initiate T4.
SR2	5 Primary	Group members have the primary role.

Task: T5 – Consult resources**Table A5. Context-based framework applied to T5**

Characteristic	T5	Details
Weight	4 High	Consulting resources has high importance.
STT1	1 No	T5 is not a scheduled task. It can occur anytime after T4.
STT2	3 Partial	T5 is partially event-triggered.
STT3	5 Two-way	Information flow is two-way between the individual and the group.
STT4	2 Text	Information flow needs to be text at least.
STT5	5 Long	T5 task lifetime is longer compared to other tasks.
STL1	1 No	Location dependency is not needed.
STL2	5 Yes	Location report is needed during T5.
STL3	5 Yes	Services are needed during T5 (e.g., Location of a book in library).
PE1	3 Medium	User expertise should be medium.
TS1	5 Yes	T5 is decomposable. It has sub tasks.
TS2	5 Yes	T5 is sequential. It follows T4.
TS3	3 Partial	T5 is partially dependant on other tasks.
TS4	2 Client-server	Communication should be available between the task executors.
TB1	5 Yes	T5 can be repeated during PBL.
TB2	1 No	Task completion/failure notification is not needed since it can be a repeated task.
TC1	5 High	T5 has high complexity since it is location dependant and services should be made available.
GSec1	2 Low	Security does not need to be high.
GSec2	2 Low	Access control does not need to be high.
GSec3	3 Medium	Data integrity is needed in T5.
GSec4	1 N/A	Confidentiality is not needed in T5.
GSec5	1 N/A	Non-repudiation is not needed in T5.
GScal	5 High	Number of nodes is high (e.g., all students in library).
GScal2	1 High	Density is high.
GScal3	1 High	Traffic is high.
GT1	5 High	System should support transparent data between the task executors. Data should be portable.
GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	1 Low	When computer is down, T5 cannot be executed.

GA2	1 Low	When network is down, T5 cannot be executed.
GA3	1 Low	When network is not available, T5 cannot be executed.
GE1	1 Low	Cost to run is low.
GE2	3 Medium	Cost for setup is medium.
GE3	3 Medium	Cost for maintenance is medium.
EP1	3 Medium	Visual distraction during T5 is medium.
EP2	3 Medium	Auditory distraction during T5 is medium.
ES1	3 Medium	Social interaction during T5 is medium between group members.
SR1	3 Individual	Individual group members initiate T5.
SR2	5 Primary	Individuals have the primary role.

Task: T6 – Detailed study of issues

Table A6. Context-based framework applied to T6

Characteristic	T6	Details
Weight	5 Very high	Detailed study of issues has very high importance.
STT1	1 No	T6 is not a scheduled task. It can occur anytime after T5
STT2	3 Partial	T6 is partially event-triggered.
STT3	3 One-way	Information flow is one-way from the source to the individual.
STT4	2 Text	Information flow needs to be text at least.
STT5	5 Long	T6 task lifetime is longer compared to other tasks.
STL1	1 No	Location dependency is not needed.
STL2	1 No	Location report is needed during T6.
STL3	1 No	Services are not needed during T6.
PE1	3 Medium	User expertise should be medium.
TS1	5 Yes	T6 is decomposable. It has sub tasks.
TS2	1 No	T6 is not sequential.
TS3	1 No	T6 is not dependant on other tasks.
TS4	1 N/A	Collaboration is not needed.
TB1	5 Yes	T6 can be repeated during PBL.
TB2	1 No	Task completion/failure notification is not needed since it can be a repeated task.
TC1	3 Medium	T6 has medium complexity.
GSec1	1 N/A	Security is not needed.
GSec2	1 N/A	Access control is not needed.
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T6.
GSec5	1 N/A	Non-repudiation is not needed in T6.
GSca1	1 Low	Number of nodes is low
GSca2	5 Low	Density is low.
GSca3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.
GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	1 Low	T6 cannot be executed if computer is down.
GA2	3 Medium	When network is down, T6 can be executed.
GA3	3 Medium	When network is not available, T6 can be executed.

GE1	2 Low	Cost to run is low. It does not require a lot of hardware/software.
GE2	2 Low	Cost to setup is low.
GE3	2 Low	Cost to maintain is low.
EP1	3 Medium	Visual distraction during T6 is medium.
EP2	3 Medium	Auditory distraction during T6 is medium.
ES1	2 Low	Social interaction during T6 is low between group members.
SR1	3 Individual	Individual group members initiate T6.
SR2	5 Primary	Individuals have the primary role.

Task: T7 – Evaluate resources

Table A7. Context-based framework applied to T7

Characteristic	T7	Details
Weight	4 High	Evaluating resources has high importance.
STT1	3 Partial	T7 is partially scheduled task. It should occur right after T6.
STT2	3 Partial	T7 is partially event-triggered. It is triggered by the completion of T6.
STT3	5 Two-way	Information flow is two-way between the individual and the group.
STT4	2 Text	Information flow needs to be text at least.
STT5	3 Medium	T7 task lifetime is medium.
STL1	3 Partial	Location dependency is partially needed. All group members do not need to be at the same place but their location should be known by the system.
STL2	3 Partial	Location report is partially needed during T7.
STL3	1 No	Services are not needed during T7.
PE1	3 Medium	User expertise should be medium.
TS1	3 Partial	T7 is partially decomposable. It has sub tasks.
TS2	5 Yes	T7 is sequential.
TS3	5 Yes	T7 is dependant on other tasks.
TS4	5 Ad-hoc	Collaboration is needed between group members.
TB1	5 Yes	T7 can be repeated during PBL.
TB2	1 No	Task completion/failure notification is not needed since it can be a repeated task.
TC1	5 High	T5 has medium complexity because of the two-way communication.
GSec1	3 Medium	Security is needed.
GSec2	3 Medium	Access control is needed.
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T7.
GSec5	3 Medium	Non-repudiation is needed in T7.
GSca1	1 Low	Number of nodes is low.
GSca2	5 Low	Density is low.
GSca3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.
GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.

GA1	5 High	When one of the computers in the network is down, T7 can still be executed.
GA2	1 Low	When network is down, T7 cannot be executed. T7 requires a two-way, ad-hoc communication.
GA3	1 Low	T7 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication between nodes in the network.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T7 is medium.
EP2	3 Medium	Auditory distraction during T7 is medium.
ES1	5 High	Social interaction during T7 is very high between group members.
SR1	5 Group	Individual group members initiate T7.
SR2	5 Primary	Group members have the primary role.

Task: T8 – Re-examine the problem**Table A8. Context-based framework applied to T8**

Characteristic	T8	Details
Weight	4 High	Re-examining the problem has high importance.
STT1	3 Partial	T8 is partially scheduled task. It should occur right after T7.
STT2	3 Partial	T8 is partially event-triggered. It is triggered by the completion of T7.
STT3	5 Two-way	Information flow is two-way between the group members.
STT4	2 Text	Information flow needs to be text at least.
STT5	3 Medium	T8 task lifetime is medium.
STL1	3 Partial	Location dependency is partially needed. All group members do not need to be at the same place but their location should be known by the system for communication.
STL2	3 Partial	Location report is partially needed during T8.
STL3	1 No	Services are not needed during T8.
PE1	3 Medium	User expertise should be medium.
TS1	3 Partial	T8 is partially decomposable. It has sub tasks.
TS2	5 Yes	T8 is sequential.
TS3	5 Yes	T8 is dependant on other tasks.
TS4	5 Ad-hoc	Collaboration is needed between group members.
TB1	5 Yes	T8 can be repeated during PBL.
TB2	1 No	Task completion/failure notification is not needed since it can be a repeated task.
TC1	5 High	T8 has medium complexity because of the two-way communication.
GSec1	3 Medium	Security is medium (to recognize group members).
GSec2	3 Medium	Access control is needed.
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T8.
GSec5	3 Medium	Non-repudiation is needed in T8.
GSca1	1 Low	Number of nodes is low.
GSca2	5 Low	Density is low.
GSca3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.

GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	5 High	When one of the computers in the network is down, T8 can still be executed.
GA2	1 Low	When network is down, T8 cannot be executed. T8 requires a two-way, ad-hoc communication.
GA3	1 Low	T8 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T8 is medium.
EP2	3 Medium	Auditory distraction during T8 is medium.
ES1	5 High	Social interaction during T8 is very high between group members.
SR1	5 Group	Group initiates T8.
SR2	5 Primary	Group members have the primary role.

Task: T9 – Revise hypothesis and learning issues

Table A9. Context-based framework applied to T9

Characteristic	T9	Details
Weight	5 Very high	Revising the hypothesis has very high importance.
STT1	5 Yes	T9 is partially scheduled task. It should occur right after T8.
STT2	5 Yes	T9 is event-triggered. It is triggered by the completion of T8.
STT3	5 Two-way	Information flow is two-way between the group members.
STT4	2 Text	Information flow needs to be text at least.
STT5	3 Medium	T9 task lifetime is medium.
STL1	3 Partial	Location dependency is partially needed. All group members do not need to be at the same place but their location should be known by the system for communication.
STL2	3 Partial	Location report is partially needed during T9.
STL3	1 No	Services are not needed during T9.
PE1	3 Medium	User expertise should be medium.
TS1	5 Yes	T9 is partially decomposable. It has sub tasks.
TS2	5 Yes	T9 is sequential.
TS3	5 Yes	T9 is dependant on other tasks.
TS4	5 Ad-hoc	Collaboration is needed between group members.
TB1	5 Yes	T9 can be repeated during PBL.
TB2	1 No	Task completion/failure notification is not needed since it can be a repeated task.
TC1	5 High	T9 has high complexity because of the two-way communication.
GSec1	3 Medium	Security is medium (to recognize group members).
GSec2	3 Medium	Access control is needed.
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T9.
GSec5	3 Medium	Non-repudiation is needed in T9. Group members need to know they are communicating with people from their own group.
GScal	1 Low	Number of nodes is low.
GScal2	5 Low	Density is low.
GScal3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.

GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	5 High	When one of the computers in the network is down, T9 can still be executed.
GA2	1 Low	When network is down, T9 cannot be executed.
GA3	1 Low	T9 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T9 is medium.
EP2	3 Medium	Auditory distraction during T9 is medium.
ES1	5 High	Social interaction during T9 is very high between group members.
SR1	5 Group	Group initiates T9.
SR2	5 Primary	Group members have the primary role.

Task: T10 – Present findings

Table A10. Context-based framework applied to T10

Characteristic	T10	Details
Weight	5 Very high	Presenting the findings has very high importance.
STT1	5 Yes	T10 is a scheduled task. It should occur right after T9 at a specific time.
STT2	5 Yes	T10 is event-triggered. It is triggered by the completion of T9.
STT3	5 Two-way	Information flow is two-way between the group members.
STT4	4 Audio	Information flow needs to be audio at least.
STT5	1 Short	T10 task lifetime is shorter compared to other tasks.
STL1	5 Yes	Location dependency is needed. Even though group members do not need to be at the same place during documentation, presentation part of the task requires same location.
STL2	5 Yes	Location report is needed during T10.
STL3	1 No	Services are not needed during T10.
PE1	3 Medium	User expertise should be at least medium.
TS1	5 Yes	T10 is decomposable. It has sub tasks (document and present best solution).
TS2	5 Yes	T10 is sequential.
TS3	5 Yes	T10 is dependant on other tasks.
TS4	4 Broadcasting	Group members need to broadcast the presentation part of the task.
TB1	1 No	T10 cannot be repeated during PBL.
TB2	5 Yes	Task completion/failure notification is needed since completion of the task is a very important step.
TC1	3 Medium	T10 has medium complexity because it requires broadcasting.
GSec1	3 Medium	Security is medium (to recognize group members, students participating in presentation session).
GSec2	3 Medium	Access control is needed and medium
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T10.
GSec5	3 Medium	Non-repudiation is needed in T10. Group members need to know they are communicating with people from their own group and people from the other groups need to know the group doing the presentation is the correct group.

GSca1	3 Medium	Number of nodes is medium (other groups joining the session for presentation).
GSca2	3 Medium	Density is medium.
GSca3	3 Medium	Traffic is medium.
GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.
GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	1 Low	When computer is down, T10 cannot be executed.
GA2	3 Medium	When network is down, task can be executed but it may not be completed.
GA3	3 Medium	When network is not available, task can be executed but it may not be completed.
GE1	2 Low	Cost is low to run T10.
GE2	3 Medium	Cost to setup T10 is medium since Audio or Video may be involved. These require high network capacity.
GE3	2 Low	Cost of maintaining the system for T10 is low.
EP1	3 Medium	Visual distraction during T10 is medium.
EP2	3 Medium	Auditory distraction during T10 is medium.
ES1	3 Medium	Social interaction during T10 is very medium between group members and other groups.
SR1	5 Group	Group initiates T10.
SR2	5 Primary	Group members have the primary role.

Task: T11- Metacognitive critique**Table A11. Context-based framework applied to T11**

Characteristic	T11	Details
Weight	5 Very high	Metacognitive critique task has very high importance.
STT1	5 Yes	T11 is a scheduled task. It should occur right after T10 at a specific time.
STT2	5 Yes	T11 is event-triggered. It is triggered by the completion of T10.
STT3	5 Two-way	Information flow is two-way between the group members, facilitator and other students.
STT4	2 Text	Information flow needs to be text at least.
STT5	3 Medium	T11 task lifetime is medium.
STL1	5 Yes	Location dependency is needed.
STL2	5 Yes	Location report is needed during T11.
STL3	1 No	Services are not needed during T11.
PE1	3 Medium	User expertise should be at least medium.
TS1	5 Yes	T11 is decomposable.
TS2	5 Yes	T11 is sequential.
TS3	5 Yes	T11 is dependant on other tasks.
TS4	5 Ad-hoc	Group members, facilitator and other students can participate in the discussion.
TB1	1 No	T11 cannot be repeated during PBL.
TB2	3 Partial	Task completion/failure notification is partially needed since completion of the task needs to be known by the participants.
TC1	5 High	T11 has high complexity because of the two-way communication.
GSec1	3 Medium	Security is medium (to recognize group members, students participating in session).
GSec2	3 Medium	Access control is needed and medium
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T11.
GSec5	3 Medium	Non-repudiation is needed in T11. Group members need to know they are communicating with people from their own group and facilitator needs to know correct students participate in session.
GScal	3 Medium	Number of nodes is medium (other groups joining the session).
GScal2	3 Medium	Density is medium.
GScal3	3 Medium	Traffic is medium.

GT1	3 Medium	System should support transparent data between the task executers. Data should be portable.
GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.
GA1	5 High	When one of the computers in the network is down, T11 can still be executed.
GA2	1 Low	When network is down, T11 cannot be executed.
GA3	1 Low	T11 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T11 is medium.
EP2	3 Medium	Auditory distraction during T11 is medium.
ES1	5 High	Social interaction during T11 is very high between group members, other groups and the facilitator.
SR1	4 Facilitator	Facilitator initiates T11.
SR2	5 Primary	Facilitator has the primary role.

Task: T12 – Reformulate further hypotheses

Table A12. Context-based framework applied to T12

Characteristic	T12	Details
Weight	3 Medium	Reformulating further hypotheses task has medium weight compared to other tasks.
STT1	3 Partial	T12 is a partially scheduled task. It should occur anytime after T11.
STT2	3 Partial	T12 is partially event-triggered. It is triggered by T11.
STT3	5 Two-way	Information flow is two-way between the group members.
STT4	2 Text	Information flow needs to be text at least.
STT5	1 Short	T12 task lifetime is short.
STL1	3 Partial	Location dependency is partially needed.
STL2	3 Partial	Location report is partially needed during T12.
STL3	1 No	Services are not needed during T12.
PE1	3 Medium	User expertise should be at least medium.
TS1	3 Partial	T12 is partially decomposable.
TS2	5 Yes	T12 is sequential.
TS3	5 Yes	T12 is dependant on other tasks.
TS4	5 Ad-hoc	Group members can participate in the discussion.
TB1	1 No	T12 cannot be repeated during PBL.
TB2	1 No	Task completion/failure notification is not needed.
TC1	3 Medium	T12 has medium complexity because of the two-way communication.
GSec1	3 Medium	Security is medium (to recognize group members).
GSec2	3 Medium	Access control is needed and medium
GSec3	1 N/A	Data integrity is not needed.
GSec4	1 N/A	Confidentiality is not needed in T12.
GSec5	3 Medium	Non-repudiation is needed in T12. Group members need to know they are communicating with people from their own group.
GSca1	1 Low	Number of nodes is low (group members).
GSca2	5 Low	Density is low.
GSca3	5 Low	Traffic is low.
GT1	3 Medium	System should support transparent data between the task executors. Data should be portable.
GT2	3 Medium	System should support transparent communication between different devices but it can be restricted to specific devices.

GA1	5 High	When one of the computers in the network is down, T12 can still be executed.
GA2	1 Low	When network is down, T12 cannot be executed.
GA3	1 Low	T12 cannot be executed when network is not available.
GE1	2 Low	Cost to run is low.
GE2	3 Medium	Cost to setup is medium since it involves two-way communication.
GE3	2 Low	Cost for maintenance is low.
EP1	3 Medium	Visual distraction during T12 is medium.
EP2	3 Medium	Auditory distraction during T12 is medium.
ES1	5 High	Social interaction during T12 is very high between group members.
SR1	5 Group	The group initiates T12.
SR2	5 Primary	Group members have the primary role.

Appendix B. Context-based Framework Software

As a supplement of this study, a web-based software program is developed to support the context-based framework process. The software automates the context-based framework process steps and displays the results from the framework application.

The software is developed using C# as the programming language on .NET platform with Microsoft SQL Server as the database component. Since it is developed specially for web, the application runs on Microsoft IIS (Internet Information Services) server on Windows 2000.

The following sections include the database structure, actual screen shots and source code of the application.

Database Component

Context-based framework stores data in a database called Framework. There are ten tables created in Framework database. These tables are:

Context

Context table is used to store context information.

Table B1. Context table

Column Name	Data Type	Length	Allow Nulls
ContextNo	int	4	
ContextName	varchar	50	
Detail	varchar	100	✓

Attribute

Attribute table is used to store attribute information.

Table B2. Attribute table

Column Name	Data Type	Length	Allow Nulls
AttributeNo	int	4	
ContextNo	int	4	
AttributeName	varchar	50	
Detail	varchar	100	✓

Characteristics

Characteristics table is used to store characteristics information.

Table B3. Characteristics table

Column Name	Data Type	Length	Allow Nulls
CharNo	int	4	
CharLabel	varchar	10	
CharName	varchar	40	
ContextNo	int	4	
AttributeNo	int	4	
CharSetNo	int	4	
Detail	varchar	100	✓

System

System table is used to store mobile projects.

Table B4. System table

Column Name	Data Type	Length	Allow Nulls
SystemNo	int	4	
SystemName	varchar	50	
Detail	varchar	150	✓
NumberOfTasks	int	4	✓

CharSetNo

CharSetNo table is used to store value sets for characteristics.

Table B5. CharSetNo table

Column Name	Data Type	Length	Allow Nulls
CharSetNo	int	4	
NumberOfValues	int	4	
SetValues	varchar	100	

Task

Task table is used to store task information.

Table B6. Task table

Column Name	Data Type	Length	Allow Nulls
TaskNo	int	4	
SystemNo	int	4	
TaskName	varchar	50	
Detail	varchar	200	✓
Weight	int	4	

ActualCharValues

ActualCharValues table is used to store the actual characteristics values for each task in the system.

Table B7. ActualCharValues table

Column Name	Data Type	Length	Allow Nulls
TaskNo	int	4	
SystemNo	int	4	
CharNo	int	4	
ActualValue	int	4	✓

SystemTargetProfileContext

SystemTargetProfileContext table is used to store the system target profile context information.

Table B8. SystemTargetProfileContext table

Column Name	Data Type	Length	Allow Nulls
SystemTargetProfileContextNo	int	4	
SystemNo	int	4	
ContextNo	int	4	
ExpectedValue	int	4	✓
ActualValue	int	4	✓

SystemTargetProfileAttribute

SystemTargetProfileAttribute table is used to store the system target profile attribute information.

Table B9. SystemTargetProfileAttribute table

Column Name	Data Type	Length	Allow Nulls
SystemTargetProfileAttributeNo	int	4	
SystemNo	int	4	
AttributeNo	int	4	
ExpectedValue	int	4	✓
ActualValue	int	4	✓

SystemTargetProfileCharacteristics

SystemTargetProfileCharacteristics table is used to store the system target profile characteristics information.

Table B10. SystemTargetProfileCharacteristics table

Column Name	Data Type	Length	Allow Nulls
SystemTargetProfileCharacteristicsNo	int	4	
SystemNo	int	4	
CharacteristicsNo	int	4	
ExpectedValue	int	4	✓
ActualValue	int	4	✓

The relationships between these tables are shown in Figure B1.

Figure B1. Table relationships

Screenshots

Figures B2-B10 show the actual flow of the context-based framework software.

Figure B2. Main page

NewProject - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail News

Address <http://localhost/ContextBasedFramework/NewProject.aspx> Go Links

Create a new project

Project No: 2

Project Name*:

of Tasks*:

Details:
max 150 chars

All fields indicated with * are required.

Done Local intranet

Figure B3. Create project page

Figure B4. Project saved page

CreateTasks - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print

Address <http://localhost/ContextBasedFramework/NewProject.aspx> Go Links

Create Tasks

Task ID	Task Name	Task Weight	Details
13	<input type="text"/>	<input type="text"/>	<input type="text"/>
14	<input type="text"/>	<input type="text"/>	<input type="text"/>
15	<input type="text"/>	<input type="text"/>	<input type="text"/>

Done Local intranet

Figure B5. Create tasks page

CreateTasks - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print

Address http://localhost/ContextBasedFramework/CreateTasks.aspx Go Links

Create Tasks

Task ID	Task Name	Task Weight	Details
13	T1	3	Task 1
14	T2	4	Task 2
15	T3	2	Task 3

Save Cancel Create SystemTarget Profile

Done Local intranet

Figure B6. Tasks saved page

SystemTargetProfile - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://localhost/ContextBasedFramework/CreateTasks.aspx Go Links

Create System Target Profile

Save

Context	Expected Value	Attribute	Expected Value	Characteristics	Expected Value
Spatio-temporal	☐ Satisfied ☐ Not Satisfied	Time	<input type="text"/>	STT1 Scheduled	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				STT2 Event-triggered	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				STT3 Information Flow Type	☐ 0 N/A ☐ 1 None ☐ 3 One-way ☐ 5 Two-way
				STT4 Information Flow	☐ 0 N/A ☐ 1 None ☐ 2 Text ☐ 3 Graphics ☐ 4 Audio ☐ 5 Video
				STT5 Task Lifetime	☐ 0 N/A ☐ 1 Short ☐ 3 Medium ☐ 5 Long
		Location	<input type="text"/>	STL1 Location Dependency	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				STL2 Location Report	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				STL3 Service Availability	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
Personal	☐ Satisfied ☐ Not Satisfied	Expertise	<input type="text"/>	PE1 User Experience Required	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
Task	☐ Satisfied ☐ Not Satisfied	Structure	<input type="text"/>	TS1 Decomposable	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				TS2 Sequential	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				TS3 Interdependant	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
				TS4 Collaboration Type	☐ 0 N/A ☐ 1 No ☐ 2 Client/Server ☐ 3 Peer-to-Peer ☐ 4 Broadcasting ☐ 5 Ad-hoc
				Behavior	<input type="text"/>
				TB2 Task completion/failure	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes

Done Local intranet

Figure B7. Create system target profile page

ActualValues - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://localhost/ContextBasedFramework/SystemTargetProfile.aspx Go Links

Actual Characteristics Values

Save

TaskName	Characteristics	Actual Value
T1	STT1 Scheduled	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	STT2 Event-triggered	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	STT3 Information Flow Type	☐ 0 N/A ☐ 1 None ☐ 3 One-way ☐ 5 Two-way
	STT4 Information Flow	☐ 0 N/A ☐ 1 None ☐ 2 Text ☐ 3 Graphics ☐ 4 Audio ☐ 5 Video
	STT5 Task Lifetime	☐ 0 N/A ☐ 1 Short ☐ 3 Medium ☐ 5 Long
	STL1 Location Dependency	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	STL2 Location Report	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	STL3 Service Availability	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	PE1 User Experience Required	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	TS1 Decomposable	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	TS2 Sequential	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	TS3 Interdependant	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	TS4 Collaboration Type	☐ 0 N/A ☐ 1 No ☐ 2 Client/Server ☐ 3 Peer-to-Peer ☐ 4 Broadcasting ☐ 5 Ad-hoc
	TB1 Repetitiveness	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
	TB2 Task completion/failure Notification	☐ 0 N/A ☐ 1 No ☐ 3 Partial ☐ 5 Yes
G	TC1 Complexity	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GSec1 Identification/Authentication	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GSec2 Authorization/Access Control	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GSec3 Data Integrity	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GSec4 Confidentiality	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GSec5 Non-repudiation	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GScal1 Number of nodes	☐ 0 N/A ☐ 1 No ☐ 2 Low ☐ 3 Medium ☐ 5 High
	GScal2 Density	☐ 0 N/A ☐ 1 No ☐ 2 High ☐ 3 Medium ☐ 5 Low

Done Local intranet

Figure B8. Add actual characteristics values page

Results - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://localhost/ContextBasedFramework/ActualValues.aspx Go Links

System Characteristics Values

Next >>

Characteristics	Expected Value	Actual Value
STT1 Scheduled	3	3.44444441795349
STT2 Event-triggered	1	3.44444441795349
STT3 Information Flow Type	5	4.11111116409302
STT4 Information Flow	3	2.44444441795349
STT5 Task Lifetime	3	1.66666662693024
STL1 Location Dependency	3	2.11111116409302
STL2 Location Report	5	3
STL3 Service Availability	5	2.33333325386047
PE1 User Experience Required	3	2.2222232818604
TS1 Decomposable	5	4.55555534362793
TS2 Sequential	3	3
TS3 Interdependent	3	3
TS4 Collaboration Type	5	4.4444445637207
TB1 Repetitiveness	3	3.66666674613953
TB2 Task completion/failure Notification	1	3.33333325386047
TC1 Complexity	3	2.7777767181396
GSec1 Identification/Authentication	5	5
GSec2 Authorization/Access Control	3	4.33333349227905
GSec3 Data Integrity	3	4.33333349227905
GSec4 Confidentiality	5	4.11111116409302
GSec5 Non-repudiation	0	3.44444441795349
GSCa1 Number of nodes	3	2.33333325386047
GSCa2 Density	3	3.44444441795349
GSCa3 Traffic	3	3.44444441795349
GT1 Data Transparency	5	3.44444441795349
GT2 Communication Transparency	5	4.11111116409302
GA1 Computer Down Time	2	1.33333337306976
GA2 Communication Network Down Time	1	1.77777779102325

Done Local intranet

Figure B9. Calculated system characteristics values

Results2 - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail News RSS

Address http://localhost/ContextBasedFramework/Results2.aspx Go Links

System Attributes Values

Attribute	Expected Value	Actual Value	Meets Expected Value
Time	3	2.44444444179535	No
Location	3	1.71794871183542	No
Expertise	2	1.98666666666667	Almost
Structure	5	4.6875	No
Behavior	3	3	Yes
Complexity	4	3.70370356241861	No
Security	5	4.72222223877907	No
Scalability	4	3.70370366838243	No
Transparency	5	3.77777779102326	No
Availability	3	2.50000002980232	No
Efficiency	2	1.46666665077209	No
Physical	1	0.592592597007751	No
Social	1	1	Yes
Role	5	4.16666671633721	No

Done Local intranet

Figure B10. Calculated system attributes values

Stored Procedures

spSelectMaxProject

```
CREATE PROCEDURE spSelectMaxProject
@ProjectNo integer output
as
select @ProjectNo= max(SystemNo) from system

if (@ProjectNo IS NULL)
    SET @ProjectNo = 1
else
    Set @ProjectNo = @ProjectNo + 1
GO
```

spInsertNewProject

```
CREATE PROCEDURE spInsertNewProject
@ProjectNo int,
@ProjectName varchar(50),
@ProjectDetails varchar(100),
@ProjectNumOfTasks int
as
insert into system values(@ProjectNo, @ProjectName, @ProjectDetails,
@ProjectNumOfTasks)
GO
```

spSelectMaxTask

```
CREATE PROCEDURE spSelectMaxTask
@TaskNo int output
as
select @TaskNo= max(TaskNo) from Task

if (@TaskNo IS NULL)
    SET @TaskNo = 1
else
    Set @TaskNo = @TaskNo + 1
GO
```

spInsertNewTask

```
CREATE PROCEDURE spInsertNewTask
@TaskNo int,
@TaskName varchar(50),
@TaskDetails varchar(200),
@TaskWeight int,
```

```

@ProjectNo int
as
insert into task values(@TaskNo, @ProjectNo, @TaskName,@TaskDetails,
@TaskWeight)
GO

```

spCreateSystemTargetProfile

```

CREATE PROCEDURE spCreateSystemTargetProfile AS

```

```

select context.contextno,context.contextname,
attribute.attributeno,attribute.attributename,characteristics.charno,
characteristics.charlabel,
charset.numberofvalues,characteristics.charname, charset.setvalues
from context, attribute, characteristics, charset
where context.contextno=attribute.contextno and
attribute.attributeno=characteristics.attributeno and
charset.charsetno = characteristics.charsetno;
GO

```

spSelectMaxSystemTargetProfileContextNo

```

CREATE PROCEDURE spSelectMaxSystemTargetProfileContextNo

```

```

@MaxContextNo integer output

```

```

as
select @MaxContextNo= max(SystemTargetProfileContextNo) from
SystemTargetProfileContext

```

```

if (@MaxContextNo IS NULL)
    SET @MaxContextNo = 1
else
    Set @MaxContextNo = @MaxContextNo + 1
GO

```

spSelectMaxSystemTargetProfileAttributeNo

```

CREATE PROCEDURE spSelectMaxSystemTargetProfileAttributeNo

```

```

@MaxAttributeNo integer output

```

```

as
select @MaxAttributeNo= max(SystemTargetProfileAttributeNo) from
SystemTargetProfileAttribute

```

```

if (@MaxAttributeNo IS NULL)
    SET @MaxAttributeNo = 1
else
    Set @MaxAttributeNo = @MaxAttributeNo + 1

```


GO

spSelectMaxSystemTargetProfileCharacteristicsNo

CREATE PROCEDURE spSelectMaxSystemTargetProfileCharacteristicsNo

@MaxCharNo integer output

as

select @MaxCharNo= max(SystemTargetProfileCharacteristicsNo) from
SystemTargetProfileCharacteristics

if (@MaxCharNo IS NULL)

SET @MaxCharNo = 1

else

Set @MaxCharNo = @MaxCharNo + 1

GO

spInsertSystemTargetProfileContext

CREATE PROCEDURE spInsertSystemTargetProfileContext

@ProfileNo int,

@ContextNo int,

@SystemNo int,

@ExpectedValue int

as

insert into SystemTargetProfileContext(SystemTargetProfileContextNo, SystemNo,
ContextNo, ExpectedValue)

values(@ProfileNo, @SystemNo, @ContextNo, @ExpectedValue)

GO

spInsertSystemTargetProfileAttribute

CREATE PROCEDURE spInsertSystemTargetProfileAttribute

@STPAttributeNo int,

@AttributeNo int,

@SystemNo int,

@ExpectedValue int

as

insert into SystemTargetProfileAttribute(SystemTargetProfileAttributeNo, SystemNo,
AttributeNo, ExpectedValue)

values(@STPAttributeNo, @SystemNo, @AttributeNo, @ExpectedValue)

GO

spInsertSystemTargetProfileCharacteristics

CREATE PROCEDURE spInsertSystemTargetProfileCharacteristics

@STPCharNo int,
@CharNo int,
@SystemNo int,
@ExpectedValue int

as

insert into SystemTargetProfileCharacteristics(SystemTargetProfileCharacteristicsNo,
SystemNo, CharacteristicsNo, ExpectedValue)

values(@STPCharNo, @SystemNo, @CharNo,@ExpectedValue)

GO

spCreateActualCharValues

CREATE PROCEDURE spCreateActualCharValues

@ProjectNo int

AS

select task.taskno,task.taskname,

characteristics.charno,characteristics.charlabel,characteristics.charname,charset.numberof
values,

charset.setvalues from characteristics, charset,task

where charset.charsetno = characteristics.charsetno and task.systemno = @ProjectNo

order by task.taskno, characteristics.charno

GO

spInsertTaskActualValues

CREATE PROCEDURE spInsertTaskActualValues

@TaskNo int,
@SystemNo int,
@CharNo int,
@ActualValue int

as

insert into actualcharvalues values(@TaskNo, @SystemNo, @CharNo, @ActualValue)

GO

spGetSystemCharValues

CREATE PROCEDURE spGetSystemCharValues

@SystemNo int

AS

```

select systemtargetprofilecharacteristics.systemno,
systemtargetprofilecharacteristics.characteristicsno,
systemtargetprofilecharacteristics.expectedvalue,
systemtargetprofilecharacteristics.actualvalue,
characteristics.attributeno, characteristics.charlabel,
characteristics.charname,
attribExp=systemtargetprofileattribute.expectedvalue

```

```

from systemtargetprofilecharacteristics,characteristics,systemtargetprofileattribute
where systemtargetprofilecharacteristics.systemno=@SystemNo and
characteristics.charno = systemtargetprofilecharacteristics.characteristicsno and
systemtargetprofileattribute.systemno = systemtargetprofilecharacteristics.systemno and
systemtargetprofileattribute.attributeno = characteristics.attributeno
GO

```

spGetTaskActual

```

CREATE PROCEDURE spGetTaskActual

```

```

@SystemNo int

```

```

AS

```

```

select distinct task.taskno, task.weight,
actualcharvalues.actualvalue, actualcharvalues.charno,
prd = (actualcharvalues.actualvalue * task.weight), system.numberoftasks

```

```

from task, actualcharvalues, system
where task.taskno=actualcharvalues.taskno and task.systemno=@SystemNo and
system.systemno = task.systemno
order by actualcharvalues.charno
GO

```

spUpdateSTPCharacteristics

```

CREATE PROCEDURE spUpdateSTPCharacteristics

```

```

@ProjectNo int,

```

```

@CharNo int,

```

```

@Actual float

```

```

AS

```

```

update SystemTargetProfileCharacteristics

```

```

set actualvalue=@Actual

```

```

where systemno=@ProjectNo and characteristicsno=@CharNo

```

```

GO

```

spGetSystemAttributeValues

```
CREATE PROCEDURE spGetSystemAttributeValues  
@SystemNo int
```

AS

```
select systemtargetprofileattribute.systemno,  
systemtargetprofileattribute.attributeno,  
systemtargetprofileattribute.expectedvalue,  
systemtargetprofileattribute.actualvalue,  
attribute.attributeno, attribute.attributename  
  
from systemtargetprofileattribute,attribute  
where systemtargetprofileattribute.systemno=@SystemNo and  
attribute.attributeno = systemtargetprofileattribute.attributeno  
GO
```

spGetSystemCharValues

```
CREATE PROCEDURE spGetSystemCharValues  
@SystemNo int
```

AS

```
select systemtargetprofilecharacteristics.systemno,  
systemtargetprofilecharacteristics.characteristicsno,  
systemtargetprofilecharacteristics.expectedvalue,  
systemtargetprofilecharacteristics.actualvalue,  
characteristics.attributeno, characteristics.charlabel,  
characteristics.charname,  
attribExp=systemtargetprofileattribute.expectedvalue  
  
from systemtargetprofilecharacteristics,characteristics,  
systemtargetprofileattribute  
where systemtargetprofilecharacteristics.systemno=@SystemNo and  
characteristics.charno = systemtargetprofilecharacteristics.characteristicsno and  
systemtargetprofileattribute.systemno = systemtargetprofilecharacteristics.systemno and  
systemtargetprofileattribute.attributeno = characteristics.attributeno  
GO
```

spUpdateSTPAttribute

```
CREATE PROCEDURE spUpdateSTPAttribute  
@ProjectNo int,  
@AttributeNo int,
```

@Actual float

AS

update SystemTargetProfileAttribute

set actualvalue=@Actual

where systemno=@ProjectNo and attributeno=@AttributeNo

GO

Source Code

MainPage.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for WebForm1.
    /// </summary>
    public class MainPage : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.Button ButtonNew;
        protected System.Web.UI.WebControls.Button ButtonOpen;
        protected System.Web.UI.WebControls.Label Label1;

        private void Page_Load(object sender, System.EventArgs e)
        {
            // Put user code to initialize the page here
        }

        #region Web Form Designer generated code
        override protected void OnInit(EventArgs e)
        {
            //
            // CODEGEN: This call is required by the ASP.NET Web
            Form Designer.
            //
            InitializeComponent();
            base.OnInit(e);
        }

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.ButtonNew.Click += new
            System.EventHandler(this.ButtonNew_Click);
            this.Load += new System.EventHandler(this.Page_Load);
        }
    }
}
```

```
    }  
    #endregion  
  
    private void ButtonNew_Click(object sender,  
        System.EventArgs e)  
    {  
  
        Response.Redirect("//localhost/ContextBasedFramework/NewProject.a  
spx");  
    }  
}
```

NewProject.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for NewProject.
    /// </summary>
    public class NewProject : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.Label Label1;
        protected System.Web.UI.WebControls.Label Label2;
        protected System.Web.UI.WebControls.Label Label3;
        protected System.Web.UI.WebControls.Label Label4;
        protected System.Web.UI.WebControls.Label lbProjectNo;
        protected System.Web.UI.WebControls.TextBox tbProjectName;
        protected System.Web.UI.WebControls.TextBox tbNumofTasks;
        protected System.Web.UI.WebControls.TextBox
tbProjectDetails;
        protected System.Web.UI.WebControls.Button ButtonSave;
        protected System.Web.UI.WebControls.Button ButtonCancel;
        protected System.Web.UI.WebControls.Label Label6;
        protected System.Web.UI.WebControls.Label Label7;
        protected System.Web.UI.WebControls.RequiredFieldValidator
reqFieldNumofTasks;
        protected System.Web.UI.WebControls.RequiredFieldValidator
reqFieldProjectName;
        protected System.Web.UI.WebControls.CompareValidator
cvNumofTasks;
        protected System.Web.UI.WebControls.Label Label5;
        protected System.Web.UI.WebControls.Label Label8;

        private SqlConnection connection;
        protected System.Web.UI.WebControls.Button
ButtonCreateTasks;
        private SqlCommand command;

        private void Page_Load(object sender, System.EventArgs e)
        {
            // Put user code to initialize the page here
            string projectNum;
            if (!IsPostBack)
```



```

        {
            projectNum = GetProjectNumber();
            lbProjectNo.Text = projectNum;
            this.Session.Add("projectNumber", projectNum);
        }
    }

private string GetProjectNumber()
{
    string projectNum = null;
    try
    {
        connection = new
        SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();
        command = new SqlCommand("spSelectMaxProject", connection);
        command.CommandType = CommandType.StoredProcedure;
        command.Parameters.Add( "@ProjectNo", SqlDbType.Int);
        command.Parameters["@ProjectNo"].Direction =
        ParameterDirection.Output;
        command.ExecuteNonQuery();

        projectNum = command.Parameters["@ProjectNo"].Value.ToString();
    }
    catch (Exception ex)
    {
        Label8.Visible = true;
        Label8.Text = ex.Message.ToString();
    }
    finally
    {
        if (connection != null)
            connection.Close();
    }

    return projectNum;
}

#region Web Form Designer generated code
override protected void OnInit(EventArgs e)
{
    //
    // CODEGEN: This call is required by the ASP.NET Web Form
    Designer.
    //
    InitializeComponent();
    base.OnInit(e);
}

/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>

```

```

        private void InitializeComponent()
        {
            this.ButtonCancel.Click += new
System.EventHandler(this.ButtonCancel_Click);
            this.ButtonSave.Click += new
System.EventHandler(this.ButtonSave_Click);
            this.ButtonCreateTasks.Click += new
System.EventHandler(this.ButtonCreateTasks_Click);
            this.Load += new System.EventHandler(this.Page_Load);

        }
    #endregion

    private void ButtonSave_Click(object sender, System.EventArgs e)
    {
        SqlConnection connection = new
        SqlConnection(Application["DBConnectionString"].ToString())
        ;

        try
        {
            connection.Open();

            SqlCommand command = new SqlCommand(
"spInsertNewProject", connection);
            command.CommandType =
CommandType.StoredProcedure;

            command.Parameters.Add(new
SqlParameter("@ProjectNo", SqlDbType.Int));
            command.Parameters["@ProjectNo"].Direction =
ParameterDirection.Input;
            command.Parameters["@ProjectNo"].Value =
Convert.ToInt32(lbProjectNo.Text);

            command.Parameters.Add(new
SqlParameter("@ProjectName", SqlDbType.VarChar, 50));
            command.Parameters["@ProjectName"].Direction =
ParameterDirection.Input;
            command.Parameters["@ProjectName"].Value =
tbProjectName.Text;

            command.Parameters.Add(new
SqlParameter("@ProjectDetails", SqlDbType.VarChar, 150));
            command.Parameters["@ProjectDetails"].Direction
= ParameterDirection.Input;
            command.Parameters["@ProjectDetails"].Value =
tbProjectDetails.Text;

            command.Parameters.Add(new
SqlParameter("@ProjectNumOfTasks", SqlDbType.Int));

            command.Parameters["@ProjectNumOfTasks"].Direction =
ParameterDirection.Input;

```

```

        command.Parameters["@ProjectNumOfTasks"].Value
= Convert.ToInt32(tbNumofTasks.Text);

        this.Session.Add("numOfTasks",tbNumofTasks.Text);

        command.ExecuteNonQuery();
        Label8.Visible = true;
        Label8.Text = "Project " + tbProjectName.Text +
" was added to the database";
        ButtonSave.Enabled = false;
        ButtonCancel.Enabled = false;
        ButtonCreateTasks.Visible = true;

    }
    catch( Exception ex )
    {
        Label8.Visible = true;
        Label8.Text = ex.Message.ToString();
    }
    finally
    {
        if (connection != null)
            connection.Close();
    }
}

private void ButtonCancel_Click(object sender,
System.EventArgs e)
{
    Response.Redirect ("//localhost/ContextBasedFramework/MainPage.asp
x");
}

private void ButtonCreateTasks_Click(object sender,
System.EventArgs e)
{
    this.Server.Transfer("CreateTasks.aspx");

    Response.Redirect ("//localhost/ContextBasedFramework/CreateTasks.
aspx");
}
}
}

```

CreateTasks.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for CreateTasks.
    /// </summary>
    public class CreateTasks : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.Label Label1;
        protected System.Web.UI.WebControls.Label Label2;
        protected System.Web.UI.WebControls.Label Label3;
        protected System.Web.UI.WebControls.Label Label4;
        protected System.Web.UI.WebControls.Label Label5;
        protected System.Web.UI.WebControls.Label Label6;
        protected System.Web.UI.WebControls.PlaceHolder
Placeholder1;

        protected System.Web.UI.WebControls.Button btSave = new
Button();
        protected System.Web.UI.WebControls.Button btCancel = new
Button();
        protected System.Web.UI.WebControls.Button
btSystemTargetProfile = new Button();

        private SqlConnection connection;
        private SqlCommand command;

        private int numOfTasks = 0;
        private int systemNo = 0;

        private void Page_Load(object sender, System.EventArgs e)
        {
            int taskNum = 0;
            if (!IsPostBack)
            {
                taskNum = GetTaskNumber();
            }

            // Put user code to initialize the page here
```

```

        int top = 0;
        numOfTasks =
Convert.ToInt32(this.Session["numOfTasks"]);
        systemNo =
Convert.ToInt32(this.Session["projectNumber"]);

        for (int i =1; i<= numOfTasks; i++)
        {
            string topValue = Convert.ToString((112 + (top
* 80))) + "px";
            string topValue2 = Convert.ToString((142 + (top
* 80))) + "px";

            TextBox tbTaskId = new TextBox();
            TextBox tbTaskName = new TextBox();
            TextBox tbTaskWeight = new TextBox();
            TextBox tbTaskDetails = new TextBox();
            RequiredFieldValidator reqFieldTaskName = new
RequiredFieldValidator();
            RequiredFieldValidator reqFieldTaskWeight = new
RequiredFieldValidator();
            CompareValidator cvTaskWeight = new
CompareValidator();

            tbTaskId.ID = "tbTaskId" + i.ToString();
            tbTaskId.ReadOnly = true;
            tbTaskId.Style["Width"] = "80px";
            tbTaskId.Style["Height"] = "24px";
            tbTaskId.Style["Position"] = "Absolute";
            tbTaskId.Style["left"] = "16px";
            tbTaskId.Style["Top"] = topValue;
            tbTaskId.Text = Convert.ToString(taskNum);

            tbTaskName.ID = "tbTaskName" + i.ToString();
            tbTaskName.MaxLength = 50;
            tbTaskName.Style["Position"] = "Absolute";
            tbTaskName.Style["left"] = "120px";
            tbTaskName.Style["Width"] = "200px";
            tbTaskName.Style["Top"] = topValue;

            reqFieldTaskName.ID = "reqFieldTaskName" +
i.ToString();
            reqFieldTaskName.ControlToValidate =
"tbTaskName" + i.ToString();
            reqFieldTaskName.ErrorMessage = "Enter task
name";
            reqFieldTaskName.Style["Position"] =
"Absolute";
            reqFieldTaskName.Style["left"] = "120px";
            reqFieldTaskName.Style["Top"] = topValue2;
            reqFieldTaskName.ForeColor = Color.Red;

            tbTaskWeight.ID = "tbTaskWeight" +

```

```

i.ToString();
    tbTaskWeight.Style["Position"] = "Absolute";
    tbTaskWeight.Style["left"] = "344px";
    tbTaskWeight.Style["Width"] = "80px";
    tbTaskWeight.Style["Top"] = topValue;

    reqFieldTaskWeight.ID = "reqFieldTaskWeight" +
i.ToString();
    reqFieldTaskWeight.ControlToValidate =
"tbTaskWeight" + i.ToString();
    reqFieldTaskWeight.ErrorMessage = "Enter task
weight";
    reqFieldTaskWeight.Style["Position"] =
"Absolute";
    reqFieldTaskWeight.Style["left"] = "344px";
    reqFieldTaskWeight.Style["Top"] = topValue2;
    reqFieldTaskWeight.ForeColor = Color.Red;

    cvTaskWeight.ID = "cvTaskWeight" +
i.ToString();
    cvTaskWeight.ControlToValidate = "tbTaskweight"
+ i.ToString();
    cvTaskWeight.ErrorMessage = "Enter an integer";
    cvTaskWeight.Operator =
ValidationCompareOperator.DataTypeCheck;
    cvTaskWeight.Type = ValidationDataType.Integer;
    cvTaskWeight.ForeColor = Color.Red;
    cvTaskWeight.Style["Position"] = "Absolute";
    cvTaskWeight.Style["left"] = "344px";
    cvTaskWeight.Style["Top"] = topValue2;

    tbTaskDetails.ID = "tbTaskDetails" +
i.ToString();
    tbTaskDetails.TextMode = TextBoxMode.MultiLine;
    tbTaskDetails.MaxLength = 200;
    tbTaskDetails.Style["Position"] = "Absolute";
    tbTaskDetails.Style["left"] = "456px";
    tbTaskDetails.Style["Width"] = "208px";
    tbTaskDetails.Style["Height"] = "64px";
    tbTaskDetails.Style["Top"] = topValue;

    Placeholder1.Controls.Add(tbTaskId);
    Placeholder1.Controls.Add(tbTaskName);
    Placeholder1.Controls.Add(tbTaskWeight);
    Placeholder1.Controls.Add(tbTaskDetails);
    Placeholder1.Controls.Add(reqFieldTaskName);
    Placeholder1.Controls.Add(reqFieldTaskWeight);
    Placeholder1.Controls.Add(cvTaskWeight);
    // Add a spacer in the form of an HTML <BR> element.
    Placeholder1.Controls.Add(new
LiteralControl("<br>"));
    top++;
    taskNum++;

```

```

    }

    btSave.Text = "Save";
    btSave.Style["position"] = "Absolute";
    btSave.Style["Height"] = "32px";
    btSave.Style["Width"] = "96px";
    btSave.Style["Left"] = "24px";
    btSave.Style["top"] = Convert.ToString((112 + (top *
80))) + "px";

    btCancel.Text = "Cancel";
    btCancel.Style["position"] = "Absolute";
    btCancel.Style["Height"] = "32px";
    btCancel.Style["Width"] = "96px";
    btCancel.Style["Left"] = "136px";
    btCancel.Style["top"] = Convert.ToString((112 + (top
* 80))) + "px";

    btCancel.CausesValidation = false;

    btSystemTargetProfile.Text = "Create SystemTarget
Profile";

    btSystemTargetProfile.Visible = false;
    btSystemTargetProfile.Style["position"] = "Absolute";
    btSystemTargetProfile.Style["Height"] = "32px";
    //btSystemTargetProfile.Style["Width"] = "96px";
    btSystemTargetProfile.Style["Left"] = "248px";
    btSystemTargetProfile.Style["top"] =
Convert.ToString((112 + (top * 80))) + "px";
    btSystemTargetProfile.CausesValidation = false;

    Placeholder1.Controls.Add(btSave);
    Placeholder1.Controls.Add(btCancel);
    Placeholder1.Controls.Add(btSystemTargetProfile);
}

private int GetTaskNumber()
{
    int taskNum = 0;
    try
    {
        connection = new
SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new SqlCommand("spSelectMaxTask",
connection);
        command.CommandType =
CommandType.StoredProcedure;
        command.Parameters.Add( "@TaskNo",
SqlDbType.Int);
        command.Parameters["@TaskNo"].Direction =
ParameterDirection.Output;
        command.ExecuteNonQuery();

```

```

        taskNum =
Convert.ToInt32(command.Parameters["@TaskNo"].Value);
    }
    catch (Exception ex)
    {
        Label6.Visible = true;
        Label6.Text = ex.Message.ToString();
    }
    finally
    {
        if (connection != null)
            connection.Close();
    }

    return taskNum;
}

#region Web Form Designer generated code
override protected void OnInit(EventArgs e)
{
    //
    // CODEGEN: This call is required by the ASP.NET Web
Form Designer.
    //
    InitializeComponent();
    base.OnInit(e);
}

/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.Load += new System.EventHandler(this.Page_Load);
    this.btSave.Click += new
System.EventHandler(this.btSave_Click);
    this.btSystemTargetProfile.Click += new
System.EventHandler(this.btSystemTargetProfile_Click);
}

private void btSave_Click(object sender, EventArgs
e)
{
    SqlConnection connection = new
SqlConnection(Application["DBConnectionString"].ToString());
    try
    {
        connection.Open();

        SqlCommand command = new SqlCommand(

```



```

"spInsertNewTask", connection);
        command.CommandType =
CommandType.StoredProcedure;

        command.Parameters.Add(new
SqlParameter("@TaskNo", SqlDbType.Int));
        command.Parameters["@TaskNo"].Direction =
ParameterDirection.Input;

        command.Parameters.Add(new
SqlParameter("@TaskName", SqlDbType.VarChar, 50));
        command.Parameters["@TaskName"].Direction =
ParameterDirection.Input;

        command.Parameters.Add(new
SqlParameter("@TaskDetails", SqlDbType.VarChar, 200));
        command.Parameters["@TaskDetails"].Direction =
ParameterDirection.Input;

        command.Parameters.Add(new
SqlParameter("@TaskWeight", SqlDbType.Int));
        command.Parameters["@TaskWeight"].Direction =
ParameterDirection.Input;

        command.Parameters.Add(new
SqlParameter("@ProjectNo", SqlDbType.Int));
        command.Parameters["@ProjectNo"].Direction =
ParameterDirection.Input;

        for (int i = 1; i <= numOfTasks; i++)
        {
            TextBox tbId =
this.Page.FindControl("tbTaskId" + i) as TextBox;
            TextBox tbName =
this.Page.FindControl("tbTaskName" + i) as TextBox;
            TextBox tbWeight =
this.Page.FindControl("tbTaskWeight" + i) as TextBox;
            TextBox tbDetails =
this.Page.FindControl("tbTaskDetails" + i) as TextBox;

            command.Parameters["@TaskNo"].Value =
Convert.ToInt32(tbId.Text);
            command.Parameters["@TaskName"].Value =
tbName.Text;
            command.Parameters["@TaskDetails"].Value =
tbDetails.Text;
            command.Parameters["@TaskWeight"].Value =
Convert.ToInt32(tbWeight.Text);
            command.Parameters["@ProjectNo"].Value =
systemNo;

            command.ExecuteNonQuery();
        }

```

```

    }
    catch( Exception ex )
    {
        Labell.Visible = true;
        Labell.Text = ex.Message.ToString();
    }
    finally
    {
        if (connection != null)
            connection.Close();
    }

    btSystemTargetProfile.Visible = true;

    //Response.Redirect("///localhost/ContextBasedFramework/CreateTask
    s.aspx");
    }

    private void btSystemTargetProfile_Click(object sender,
    System.EventArgs e)
    {
        this.Session.Add("systemNo", systemNo);
        this.Server.Transfer("SystemTargetProfile.aspx");

        Response.Redirect("///localhost/ContextBasedFramework/SystemTarget
        Profile.aspx");
    }
    #endregion
}
}

```

SystemTargetProfile.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for SystemTargetProfile.
    /// </summary>
    public class SystemTargetProfile : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.DataGrid DataGrid1;
        protected System.Web.UI.WebControls.Label Label1;

        private SqlConnection connection;
        private SqlCommand command;
        private static string prevContext = null;
        private static string prevAttrib = null;
        private static string prevExpected = null;
        protected System.Web.UI.WebControls.Button Button1;
        protected System.Web.UI.WebControls.Button btSave;
        private static int prevAttribExpected = 0;
        private int systemNo = 0;

        private void Page_Load(object sender, System.EventArgs e)
        {
            systemNo = Convert.ToInt32(this.Session["systemNo"]);
            if (!IsPostBack)
            {
                BindData();
            }
            // Put user code to initialize the page here
        }

        private void BindData()
        {
            DataGrid1.DataSource = GetSystemTargetProfile();
            DataGrid1.DataBind();
        }

        private SqlDataReader GetSystemTargetProfile()
        {
            connection = new
```

```

SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new
SqlCommand("spCreateSystemTargetProfile", connection);
        command.CommandType = CommandType.StoredProcedure;

        return
command.ExecuteReader(CommandBehavior.CloseConnection);
    }

    private int GetSTPContextNumber()
    {
        int contextNum = 0;
        try
        {
            connection = new
SqlConnection(Application["DBConnectionString"].ToString());
            connection.Open();

            command = new
SqlCommand("spSelectMaxSystemTargetProfileContextNo", connection);
            command.CommandType =
CommandType.StoredProcedure;
            command.Parameters.Add( "@MaxContextNo",
SqlDbType.Int);
            command.Parameters["@MaxContextNo"].Direction =
ParameterDirection.Output;
            command.ExecuteNonQuery();

            contextNum =
Convert.ToInt32(command.Parameters["@MaxContextNo"].Value);
        }
        catch (Exception ex)
        {
            Labell.Text = ex.Message.ToString();
        }
        finally
        {
            if (connection != null)
                connection.Close();
        }

        return contextNum;
    }

    private int GetSTPAttributeNumber()
    {
        int attributeNum = 0;
        try
        {
            connection = new

```

```

SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new
SqlCommand("spSelectMaxSystemTargetProfileAttributeNo", connection);
        command.CommandType =
CommandType.StoredProcedure;
        command.Parameters.Add( "@MaxAttributeNo",
SqlDbType.Int);
        command.Parameters["@MaxAttributeNo"].Direction
= ParameterDirection.Output;
        command.ExecuteNonQuery();

        attributeNum =
Convert.ToInt32(command.Parameters["@MaxAttributeNo"].Value);
    }
    catch (Exception ex)
    {
        Labell1.Text = ex.Message.ToString();
    }
    finally
    {
        if (connection != null)
            connection.Close();
    }

    return attributeNum;
}

private int GetSTPCharNumber()
{
    int charNum = 0;
    try
    {
        connection = new
SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new
SqlCommand("spSelectMaxSystemTargetProfileCharacteristicsNo",
connection);
        command.CommandType =
CommandType.StoredProcedure;
        command.Parameters.Add( "@MaxCharNo",
SqlDbType.Int);
        command.Parameters["@MaxCharNo"].Direction =
ParameterDirection.Output;
        command.ExecuteNonQuery();

        charNum =
Convert.ToInt32(command.Parameters["@MaxCharNo"].Value);
    }
    catch (Exception ex)

```

```

        {
            Label1.Text = ex.Message.ToString();
        }
        finally
        {
            if (connection != null)
                connection.Close();
        }

        return charNum;
    }

    private void SaveSystemTargetProfile()
    {
        int contextNo = GetSTPContextNumber();
        int attributeNo = GetSTPAttributeNumber();
        int charNo = GetSTPCharNumber();
        SqlConnection connection = new
        SqlConnection(Application["DBConnectionString"].ToString());
        try
        {
            connection.Open();

            SqlCommand command = new SqlCommand(
            "spInsertSystemTargetProfileContext", connection);
            command.CommandType =
            CommandType.StoredProcedure;

            command.Parameters.Add(new
            SqlParameter("@ProfileNo", SqlDbType.Int));

            command.Parameters["@ProfileNo"].Direction =
            ParameterDirection.Input;

            command.Parameters.Add(new
            SqlParameter("@ContextNo", SqlDbType.Int));

            command.Parameters["@ContextNo"].Direction =
            ParameterDirection.Input;

            command.Parameters.Add(new
            SqlParameter("@SystemNo", SqlDbType.Int));
            command.Parameters["@SystemNo"].Direction
            = ParameterDirection.Input;

            command.Parameters.Add(new
            SqlParameter("@ExpectedValue", SqlDbType.Int));

            command.Parameters["@ExpectedValue"].Direction =
            ParameterDirection.Input;
        }
    }

```

```

        SqlCommand command2 = new SqlCommand(
"spInsertSystemTargetProfileAttribute", connection);
        command2.CommandType =
CommandType.StoredProcedure;

        command2.Parameters.Add(new
SqlParameter("@STPAttributeNo", SqlDbType.Int));

        command2.Parameters["@STPAttributeNo"].Direction =
ParameterDirection.Input;

        command2.Parameters.Add(new
SqlParameter("@AttributeNo", SqlDbType.Int));

        command2.Parameters["@AttributeNo"].Direction =
ParameterDirection.Input;

        command2.Parameters.Add(new
SqlParameter("@SystemNo", SqlDbType.Int));

        command2.Parameters["@SystemNo"].Direction =
ParameterDirection.Input;

        command2.Parameters.Add(new
SqlParameter("@ExpectedValue", SqlDbType.Int));

        command2.Parameters["@ExpectedValue"].Direction =
ParameterDirection.Input;

        SqlCommand command3 = new SqlCommand(
"spInsertSystemTargetProfileCharacteristics", connection);
        command3.CommandType =
CommandType.StoredProcedure;

        command3.Parameters.Add(new
SqlParameter("@STPCharNo", SqlDbType.Int));

        command3.Parameters["@STPCharNo"].Direction =
ParameterDirection.Input;

        command3.Parameters.Add(new
SqlParameter("@CharNo", SqlDbType.Int));
        command3.Parameters["@CharNo"].Direction
= ParameterDirection.Input;

        command3.Parameters.Add(new
SqlParameter("@SystemNo", SqlDbType.Int));

        command3.Parameters["@SystemNo"].Direction =
ParameterDirection.Input;

        command3.Parameters.Add(new

```

```

SqlParameter("@ExpectedValue", SqlDbType.Int));

    command3.Parameters["@ExpectedValue"].Direction =
ParameterDirection.Input;

                                //iterate through Data Grid rows
                                foreach(DataGridItem dataGridItem in
DataGrid1.Items)
                                {

                                    //Add context information for
SystemTargetProfile
                                    if
((Label)dataGridItem.FindControl("lbContext")).Visible == true)
                                    {
                                        String context=
((Label)dataGridItem.FindControl("lbContextNo")).Text.Trim();
                                        int contextExpectedValue =
Convert.ToInt32(((RadioButtonList)dataGridItem.FindControl("ContextExpe
ctedValue")).SelectedItem.Value);

                                        command.Parameters["@ProfileNo"].Value = contextNo;

                                        command.Parameters["@ContextNo"].Value =
Convert.ToString(context);

                                        command.Parameters["@SystemNo"].Value = systemNo;

                                        command.Parameters["@ExpectedValue"].Value =
contextExpectedValue;

                                        command.ExecuteNonQuery();
                                        contextNo++;
                                    }

                                    //Add attribute information for
SystemTargetProfile
                                    if
((Label)dataGridItem.FindControl("lbAttribute")).Visible == true)
                                    {
                                        String attribute=
((Label)dataGridItem.FindControl("lbAttributeNo")).Text.Trim();
                                        int AttribExpectedValue =
Convert.ToInt32(((TextBox)dataGridItem.FindControl("AttributeExpectedVa
lue")).Text);

                                        command2.Parameters["@STPAttributeNo"].Value = attributeNo;

                                        command2.Parameters["@AttributeNo"].Value =
Convert.ToString(attribute);

```



```

        command2.Parameters["@SystemNo"].Value = systemNo;

        command2.Parameters["@ExpectedValue"].Value =
AttribExpectedValue;

                                command2.ExecuteNonQuery();
                                attributeNo++;
                                }

                                //Add characteristics information
for SystemTargetProfile

                                string charExpected = null;
                                int expected = 0;

                                if
((RadioButton)dataGridItem.FindControl("RadioButton1")).Checked ==
true)
                                charExpected =
((RadioButton)dataGridItem.FindControl("RadioButton1")).Text;
                                if
((RadioButton)dataGridItem.FindControl("RadioButton2")).Checked ==
true)
                                charExpected =
((RadioButton)dataGridItem.FindControl("RadioButton2")).Text;
                                if
((RadioButton)dataGridItem.FindControl("RadioButton3")).Checked ==
true)
                                charExpected =
((RadioButton)dataGridItem.FindControl("RadioButton3")).Text;
                                if
((RadioButton)dataGridItem.FindControl("RadioButton4")).Checked ==
true)
                                charExpected =
((RadioButton)dataGridItem.FindControl("RadioButton4")).Text;
                                if
((RadioButton)dataGridItem.FindControl("RadioButton5")).Checked ==
true &&
((RadioButton)dataGridItem.FindControl("RadioButton5")).Visible
== true)
                                charExpected =
((RadioButton)dataGridItem.FindControl("RadioButton5")).Text;
                                if
((RadioButton)dataGridItem.FindControl("RadioButton6")).Checked ==
true &&
((RadioButton)dataGridItem.FindControl("RadioButton6")).Visible
== true)
                                charExpected =
((RadioButton)dataGridItem.FindControl("RadioButton6")).Text;
                                if (charExpected != null)
                                expected =

```

```

Convert.ToInt32(charExpected.Substring(0,1));

        String chars=
((Label)dataGridItem.FindControl("lbCharNo")).Text.Trim();

        command3.Parameters["@STPCharNo"].Value = charNo;
        command3.Parameters["@CharNo"].Value = Convert.ToInt32(chars);
        command3.Parameters["@SystemNo"].Value = systemNo;
        command3.Parameters["@ExpectedValue"].Value = expected;

        command3.ExecuteNonQuery();

        charNo++;
    }

}

catch( Exception ex )
{
    Labell1.Text = ex.Message.ToString();
}
finally
{
    if (connection != null)
        connection.Close();
}
}

protected bool DisplayContext(string curContext)
{
    if (prevContext == curContext)
    {
        prevContext = curContext;
        return false;
    }
    else
    {
        prevContext = curContext;
        return true;
    }
}

protected bool DisplayAttribute(string curAttrib)
{
    if (prevAttrib == curAttrib)
    {
        prevAttrib = curAttrib;

```

```

        return false;
    }
    else
    {
        prevAttrib = curAttrib;
        return true;
    }
}

protected bool DisplayExpectedValue(string curExpected)
{
    if (prevExpected == curExpected)
    {
        prevExpected = curExpected;
        return false;
    }
    else
    {
        prevExpected = curExpected;
        return true;
    }
}

protected bool DisplayAttributeExpected(int
curAttribExpected)
{
    if (prevAttribExpected == curAttribExpected)
    {
        prevAttribExpected = curAttribExpected;
        return false;
    }
    else
    {
        prevAttribExpected = curAttribExpected;
        return true;
    }
}

public bool ShowRadioButton5(int numOfValues)
{
    if (numOfValues == 5 || numOfValues == 6)
        return true;
    else
        return false;
}

public bool ShowRadioButton6(int numOfValues)
{
    if (numOfValues == 6)
        return true;
    else
        return false;
}

```

```

        public string SplitString(string stringToSplit, int
element, string numOfValues)
        {
            Array values = stringToSplit.Split(',');

            if (element <= Convert.ToInt32(numOfValues))
                return (values.GetValue(element-1).ToString());
            else
                return null;
        }

#region Web Form Designer generated code
override protected void OnInit(EventArgs e)
{
    //
    // CODEGEN: This call is required by the ASP.NET Web
Form Designer.
    //
    InitializeComponent();
    base.OnInit(e);
}

/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.btSave.Click += new
System.EventHandler(this.btSave_Click);
    this.Load += new System.EventHandler(this.Page_Load);
}
#endregion

private void btSave_Click(object sender, EventArgs
e)
{
    SaveSystemTargetProfile();
    this.Session.Add("systemNo", systemNo);
    this.Server.Transfer("ActualValues.aspx");

    Response.Redirect("//localhost/ContextBasedFramework/ActualValues
.aspx");
}

}

```

ActualValues.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for ActualValues.
    /// </summary>
    public class ActualValues : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.DataGrid DataGrid1;
        protected System.Web.UI.WebControls.Label Label1;

        private SqlConnection connection;
        private SqlCommand command;
        private int systemNo = 0;
        protected System.Web.UI.WebControls.Button btnSave;
        private static string prevTask = null;

        private void Page_Load(object sender, System.EventArgs e)
        {
            // Put user code to initialize the page here
            systemNo = Convert.ToInt32(this.Session["systemNo"]);
            if (!IsPostBack)
            {
                BindData();
            }
        }

        private void BindData()
        {
            DataGrid1.DataSource = GetSystemTasks();
            DataGrid1.DataBind();
        }

        private SqlDataReader GetSystemTasks()
        {
            connection = new
SqlConnection(Application["DBConnectionString"].ToString());
            connection.Open();

            command = new SqlCommand("spCreateActualCharValues",
```

```

connection);

        command.CommandType = CommandType.StoredProcedure;

        command.Parameters.Add("@ProjectNo", SqlDbType.Int);
        command.Parameters["@ProjectNo"].Direction =
ParameterDirection.Input;
        command.Parameters["@ProjectNo"].Value = systemNo;

        return
command.ExecuteReader(CommandBehavior.CloseConnection);

    }

    public bool ShowRadioButton5(int numOfValues)
    {
        if (numOfValues == 5 || numOfValues == 6)
            return true;
        else
            return false;
    }

    public bool ShowRadioButton6(int numOfValues)
    {
        if (numOfValues == 6)
            return true;
        else
            return false;
    }

    public string SplitString(string stringToSplit, int
element, string numOfValues)
    {
        Array values = stringToSplit.Split(',');

        if (element <= Convert.ToInt32(numOfValues))
            return (values.GetValue(element-1).ToString());
        else
            return null;
    }

    protected bool DisplayTaskName(string curTask)
    {
        if (prevTask == curTask)
        {
            prevTask = curTask;
            return false;
        }
        else
        {
            prevTask = curTask;
            return true;
        }
    }

```

```

    }

    private void SaveActualValues()
    {
        SqlConnection connection = new
        SqlConnection(Application["DBConnectionString"].ToString());
        try
        {
            connection.Open();

            SqlCommand command = new SqlCommand(
            "spInsertTaskActualValues", connection);
            command.CommandType =
            CommandType.StoredProcedure;

            command.Parameters.Add(new
            SqlParameter("@TaskNo", SqlDbType.Int));
            command.Parameters["@TaskNo"].Direction =
            ParameterDirection.Input;

            command.Parameters.Add(new
            SqlParameter("@CharNo", SqlDbType.Int));
            command.Parameters["@CharNo"].Direction =
            ParameterDirection.Input;

            command.Parameters.Add(new
            SqlParameter("@SystemNo", SqlDbType.Int));
            command.Parameters["@SystemNo"].Direction =
            ParameterDirection.Input;

            command.Parameters.Add(new
            SqlParameter("@ActualValue", SqlDbType.Int));
            command.Parameters["@ActualValue"].Direction =
            ParameterDirection.Input;

            //iterate through Data Grid rows
            foreach(DataGridItem dataGridItem in
            DataGrid1.Items)
            {
                //Add characteristics information for
                Actual Values

                string charActual = null;
                int actual = 0;

                if
                (((RadioButton)dataGridItem.FindControl("RadioButton1")).Checked ==
                true)
                    charActual =
                    ((RadioButton)dataGridItem.FindControl("RadioButton1")).Text;
                if
                (((RadioButton)dataGridItem.FindControl("RadioButton2")).Checked ==
                true)
                    charActual =

```

```

        ((RadioButton)dataGridItem.FindControl("RadioButton2")).Text;
        if
        (((RadioButton)dataGridItem.FindControl("RadioButton3")).Checked ==
true)
            charActual =
        ((RadioButton)dataGridItem.FindControl("RadioButton3")).Text;
        if
        (((RadioButton)dataGridItem.FindControl("RadioButton4")).Checked ==
true)
            charActual =
        ((RadioButton)dataGridItem.FindControl("RadioButton4")).Text;
        if
        (((RadioButton)dataGridItem.FindControl("RadioButton5")).Checked ==
true &&
        ((RadioButton)dataGridItem.FindControl("RadioButton5")).Visible
== true)
            charActual =
        ((RadioButton)dataGridItem.FindControl("RadioButton5")).Text;
        if
        (((RadioButton)dataGridItem.FindControl("RadioButton6")).Checked ==
true &&
        ((RadioButton)dataGridItem.FindControl("RadioButton6")).Visible
== true)
            charActual =
        ((RadioButton)dataGridItem.FindControl("RadioButton6")).Text;
        if (charActual != null)
            actual =
Convert.ToInt32(charActual.Substring(0,1));

        String chars=
        ((Label)dataGridItem.FindControl("lbCharNo")).Text.Trim();
        String task=
        ((Label)dataGridItem.FindControl("lbTaskNo")).Text.Trim();

        command.Parameters["@TaskNo"].Value =
Convert.ToInt32(task);
        command.Parameters["@CharNo"].Value =
Convert.ToInt32(chars);
        systemNo;
        command.Parameters["@SystemNo"].Value =
        command.Parameters["@ActualValue"].Value
= actual;

        command.ExecuteNonQuery();
    }
}
catch( Exception ex )
{
    Labell1.Text = ex.Message.ToString();
}

```



```

        finally
        {
            if (connection != null)
                connection.Close();
        }

    }

    #region Web Form Designer generated code
    override protected void OnInit(EventArgs e)
    {
        //
        // CODEGEN: This call is required by the ASP.NET Web
Form Designer.
        //
        InitializeComponent();
        base.OnInit(e);
    }

    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.btSave.Click += new
System.EventHandler(this.btSave_Click);
        this.Load += new System.EventHandler(this.Page_Load);

    }
    #endregion

    private void btSave_Click(object sender, EventArgs
e)
    {
        SaveActualValues();
        this.Session.Add("systemNo", systemNo);
        this.Server.Transfer("Results.aspx");

        Response.Redirect("///localhost/ContextBasedFramework/Results.aspx
");
    }

}

```

Results.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for Results.
    /// </summary>
    public class Results : System.Web.UI.Page
    {
        private SqlConnection connection;
        private SqlCommand command;
        private int systemNo = 0;
        protected System.Web.UI.WebControls.Label Label1;
        protected System.Web.UI.WebControls.DataGrid DataGrid1;
        protected System.Web.UI.WebControls.Button btnNext;

        private void Page_Load(object sender, System.EventArgs e)
        {
            // Put user code to initialize the page here
            systemNo = Convert.ToInt32(this.Session["systemNo"]);
            GetSystemCharActualValues();
            if (!IsPostBack)
            {
                BindData();
            }
        }

        private void BindData()
        {
            DataGrid1.DataSource = GetSystemCharValues();
            DataGrid1.DataBind();
        }

        private SqlDataReader GetSystemCharValues()
        {
            connection = new
            SqlConnection(Application["DBConnectionString"].ToString());
            connection.Open();
        }
    }
}
```

```

        connection);
        command = new SqlCommand("spGetSystemCharValues",
        command.CommandType = CommandType.StoredProcedure;

        command.Parameters.Add("@SystemNo", SqlDbType.Int);
        command.Parameters["@SystemNo"].Direction =
ParameterDirection.Input;
        command.Parameters["@SystemNo"].Value = systemNo;

        return
        command.ExecuteReader(CommandBehavior.CloseConnection);
    }

```

```

    private void GetSystemCharActualValues()
    {
        int numTask = 0;
        int totalChar = 37;
        int[] productArray = new int[totalChar];
        float[] actualValues = new float[totalChar];
        int taskWeight = 0;
        int count = 0;
        int i = 0;

        connection = new
SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new SqlCommand("spGetTaskActual",
connection);
        command.CommandType = CommandType.StoredProcedure;

        command.Parameters.Add("@SystemNo", SqlDbType.Int);
        command.Parameters["@SystemNo"].Direction =
ParameterDirection.Input;
        command.Parameters["@SystemNo"].Value = systemNo;

        SqlDataReader rdr = command.ExecuteReader();
        while (rdr.Read())
        {
            numTask =
Convert.ToInt32(rdr["numberoftasks"].ToString());
            if (count < numTask)
            {
                productArray[i] = productArray[i] +
Convert.ToInt32(rdr["prd"].ToString());
                count++;
            }
            else
            {
                i++;
                productArray[i] =
Convert.ToInt32(rdr["prd"].ToString());

```

```

        count = 1;
    }

    taskWeight = taskWeight +
Convert.ToInt32(rdr["weight"].ToString());

    }

    rdr.Close();

    taskWeight = taskWeight / totalChar;

    for (int j=0; j<totalChar; j++)
    {
        actualValues[j] =
(float)productArray[j]/(float)taskWeight;
    }

    connection = new
SqlConnection(Application["DBConnectionString"].ToString());
    connection.Open();

    command = new
SqlCommand("spUpdateSTPCharacteristics", connection);
    command.CommandType = CommandType.StoredProcedure;

    command.Parameters.Add("@ProjectNo", SqlDbType.Int);
    command.Parameters["@ProjectNo"].Direction =
ParameterDirection.Input;

    command.Parameters.Add("@Actual", SqlDbType.Float);
    command.Parameters["@Actual"].Direction =
ParameterDirection.Input;

    command.Parameters.Add("@CharNo", SqlDbType.Int);
    command.Parameters["@CharNo"].Direction =
ParameterDirection.Input;

    for (int l=0;l<actualValues.Length;l++)
    {
        command.Parameters["@ProjectNo"].Value =
systemNo;

        command.Parameters["@CharNo"].Value = l+1;
        command.Parameters["@Actual"].Value =
actualValues[l];

        command.ExecuteNonQuery();
    }

    if (connection != null)
        connection.Close();
}

```

```

#region Web Form Designer generated code
override protected void OnInit(EventArgs e)
{
    //
    // CODEGEN: This call is required by the ASP.NET Web
Form Designer.
    //
    InitializeComponent();
    base.OnInit(e);
}

/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.btNext.Click += new
System.EventHandler(this.btNext_Click);
    this.Load += new System.EventHandler(this.Page_Load);
}
#endregion

private void btNext_Click(object sender, EventArgs
e)
{
    this.Session.Add("systemNo", systemNo);
    this.Server.Transfer("Results2.aspx");

    Response.Redirect("//localhost/ContextBasedFramework/Results2.asp
x");
}
}

```

Results2.aspx.cs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

namespace ContextBasedFramework
{
    /// <summary>
    /// Summary description for Results2.
    /// </summary>
    ///

    public class Results2 : System.Web.UI.Page
    {
        protected System.Web.UI.WebControls.Label Label1;

        private SqlConnection connection;
        private SqlCommand command;
        protected System.Web.UI.WebControls.DataGrid DataGrid1;
        private int systemNo = 0;

        public class attribData
        {
            public int attribNumber;
            public double[] charActual;
            public double[] charExpected;
            public int attribExpected;

            public attribData()
            {
                charActual = new double[10]{-1,-1,-1,-1,-1,-1,-1,-1,-1,-1};
                charExpected = new double[10]{-1,-1,-1,-1,-1,-1,-1,-1,-1,-1};
            }
        };

        private void Page_Load(object sender, System.EventArgs e)
        {
```

```

        // Put user code to initialize the page here
        systemNo = Convert.ToInt32(this.Session["systemNo"]);
        GetSystemValues();
        if (!IsPostBack)
        {
            BindData();
        }
    }

    private void BindData()
    {
        DataGrid1.DataSource = GetSystemAttribValues();
        DataGrid1.DataBind();
    }

    private SqlDataReader GetSystemAttribValues()
    {
        connection = new
        SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new
        SqlCommand("spGetSystemAttributeValues", connection);
        command.CommandType = CommandType.StoredProcedure;

        command.Parameters.Add("@SystemNo", SqlDbType.Int);
        command.Parameters["@SystemNo"].Direction =
        ParameterDirection.Input;
        command.Parameters["@SystemNo"].Value = 2;
        //systemNo;

        return
        command.ExecuteReader(CommandBehavior.CloseConnection);
    }

    private void GetSystemValues()
    {
        int attribNo = 0;

        double charActual = 0.0;
        double charExpected = 0.0;
        int attribExpected = 0;
        float[] actualAttrib = new float[14];

        connection = new
        SqlConnection(Application["DBConnectionString"].ToString());
        connection.Open();

        command = new SqlCommand("spGetSystemCharValues",
        connection);
        command.CommandType = CommandType.StoredProcedure;

```

```

        command.Parameters.Add("@SystemNo", SqlDbType.Int);
        command.Parameters["@SystemNo"].Direction =
ParameterDirection.Input;
        command.Parameters["@SystemNo"].Value = 2;//systemNo;

        SqlDataReader rdr = command.ExecuteReader();

        attribData[] adList = new attribData[14];
        for (int j=0;j<14;j++)
        {
            adList[j] = new attribData();
        }
        int count = 0;
        int prevAttrib = 0;

        while (rdr.Read())
        {
            attribData ad; //= new attribData();

            attribNo=
.Convert.ToInt32(rdr["attributeno"].ToString());
            charActual =
.Convert.ToDouble(rdr["actualvalue"].ToString());
            charExpected =
.Convert.ToDouble(rdr["expectedvalue"].ToString());
            attribExpected =
.Convert.ToInt32(rdr["attribexp"].ToString());

            if (attribNo > prevAttrib)
                count = 0;
            ad = adList[attribNo-1];
            ad.attribNumber = attribNo;
            ad.charExpected[count] = charExpected;
            ad.charActual[count] = charActual;
            ad.attribExpected = attribExpected;
            count++;
            prevAttrib = attribNo;
        }

        rdr.Close();
        double [] calcAttribute = new double[14];
        for (int kk=0 ; kk < 14; kk++)
        {
            attribData ad = adList[kk];
            int cnt = 0;
            double expectedSum = 0.0;
            calcAttribute[kk] = 0.0;
            while ( ad.charExpected[cnt] != -1)
            {
                expectedSum += ad.charExpected[cnt];
                cnt++;
            }
            cnt = 0;

```



```

        while(ad.charActual[cnt] != -1)
        {
            double actual = ad.charActual[cnt];
            double expect = ad.charExpected[cnt];
            double weight = (expect <= actual) ? 1 :
(actual/expect);

            calcAttribute[kk] += weight * expect /
expectedSum ;

            cnt++;
        }

        calcAttribute[kk] *= ad.attribExpected;
    }

    connection = new
SqlConnection(Application["DBConnectionString"].ToString());
    connection.Open();

    command = new SqlCommand("spUpdateSTPAttribute",
connection);
    command.CommandType = CommandType.StoredProcedure;

    command.Parameters.Add("@ProjectNo", SqlDbType.Int);
    command.Parameters["@ProjectNo"].Direction =
ParameterDirection.Input;

    command.Parameters.Add("@Actual", SqlDbType.Float);
    command.Parameters["@Actual"].Direction =
ParameterDirection.Input;

    command.Parameters.Add("@AttributeNo",
SqlDbType.Int);
    command.Parameters["@AttributeNo"].Direction =
ParameterDirection.Input;

    for (int l=0;l<calcAttribute.Length;l++)
    {
        command.Parameters["@ProjectNo"].Value =
2;//systemNo;

        command.Parameters["@AttributeNo"].Value = l+1;
        command.Parameters["@Actual"].Value =
calcAttribute[l];

        command.ExecuteNonQuery();
    }

    if (connection != null)
        connection.Close();
}

public Color selectColor(string actual, string expected)
{
    double actual1 = Convert.ToDouble(actual);

```

```

        double expected1 = Convert.ToDouble(expected);

        double difference = expected1 - actual1;

        if (difference == 0)
            return Color.LightGreen;
        if (difference > 0.2)
            return Color.Red;
        if (difference <= 0.2)
            return Color.Yellow;

        return Color.PowderBlue;
    }

    public string selectText(string actual, string expected)
    {
        double actual1 = Convert.ToDouble(actual);
        double expected1 = Convert.ToDouble(expected);

        double difference = expected1 - actual1;
        if (difference == 0)
            return "Yes";
        if (difference > 0.2)
            return "No";
        if (difference <= 0.2)
            return "Almost";

        return "Error";
    }

    #region Web Form Designer generated code
    override protected void OnInit(EventArgs e)
    {
        //
        // CODEGEN: This call is required by the ASP.NET Web
Form Designer.
        //
        InitializeComponent();
        base.OnInit(e);
    }

    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.Load += new System.EventHandler(this.Page_Load);
    }

    #endregion
}

```