Scientific
Research

Re-Evaluating Media Richness Theory in Software Development Settings

Mohammed A. Bindrees, Robert J. Pooley, Idris S. Ibrahim, Nick K. Taylor

School of Maths and Computer Sciences, Heriot-Watt University, Edinburgh, UK
Email: mb336@hw.ac.uk, R.J.Pooley@hw.ac.uk, I.S.Ibrahim@hw.ac.uk, N.K.Taylor@hw.ac.uk

Abstract

Software development teams communicate differently by using a variety of communication tools. Successful communication leads to competitive software based on clear and quickly delivered requirements, as well as smoothness in bug reporting and explanation. Agile and Waterfall software development approaches have both addressed the importance of communication for their process. However, neither Agile nor Waterfall has guaranteed communication effectiveness during their development lifecycle. In this study we highlight the main differences between Agile and Waterfall approaches in the light of Media Richness Theory (MRT). We also identify the preferred communication tools during a project's lifecycle using both Agile and Waterfall models separately. A mixed-method approach was employed in this study incorporating quantitative and qualitative data from interviews and a multilingual web-based survey. The results are presented descriptively and statistically and a rank ordering of communication tools based on our participants' preferences leads to a better understanding of how to select the best tool in a given situation. Thus a new updated MRT ranking model tailored for software development environment was developed, as well as, we conclude that communication tools are employed differently based on project stages and team member's role. These differences in using communication tools could be also attributed to the type of transferable information or personal preferences.

Keywords

1. Introduction

Interaction in software development workplaces is inevitably important in order to deliver the Software projects properly, and hence, communication skills have become crucial in bridging interaction's activities. Sufficiently

effective communication throughout software development stages is crucial to avoid project failure [1] [2].

Literally, communication use in software development is one of the observed reasons of project's failure from the perspective of software developers [3], although communication's difficulties increase in offshore development projects (outsourcing development) as immediate communication within different time zone and geographical locations [4].

In this paper, we investigate the factors that might influence communication tools selectivity in software development environments, and how important these factors to shape our understanding of communication obstacles. We start this research by giving a background on the importance of communication and coordination, then a study model is introduced and explained, followed by the theoretical framework, then a literature comparative between Agile and waterfall approaches in terms of communication tools' applicability, then research methodology and results is explained thoroughly, discussion of the results led to this paper conclusion and final model.

2. Background

Communication is conceptualised and supported practically within projects by the Project Management Body of Knowledge (PMBOK) as the Knowledge Area that employs the processes required to ensure timely and appropriate generation, collection, distribution, storage, retrieval, and ultimate disposition of project information [5]. The first commonly used communication model was introduced by Shannon and Weaver in 1949 [6]; this model consists of five components: Information source, Transmitter, Channel, Receiver, and Destination. A noise source appears in Shannon's model as interference with the channel component, which may cause some misinterpreting of the message. Although this model illustrates communication in a simple way, it has been criticized with regard to communication directions and dynamics (it is not a recursive model).

Furthermore, improved project coordination can be achieved by effective communication methods [1] [7]-[9]. As we expected, using face-to-face communication was the dominant communication tool in many studies [1] [2] [10]-[17]. However, the selection of communication tools is influenced by many factors such as: uncertainty, complexity of the task, and the availability of the selected tools as well as the familiarity in using these tools [18]. Several theories have described the selection process for communication tools. Media Richness Theory (MRT) identifies a positive correlation between the task uncertainty and the richness of communication tools; if the task is uncertain or more complicated, then a richer medium is needed [19]. Social Presence theory also places more emphasis on the communication tools creating togetherness between communication parties, the best tools can lead to higher togetherness and hence better communication [20]. Media Synchronicity theory uses the knowledge management concept to create an efficient communication process by emphasising the sharing of information and the building of common understanding amongst communication parties [21].

3. Study Model

Practically, Software development team members need to be easy communicated and coordinated throughout project's lifecycle in order to deliver cohesive software. However, communication channels could vary, or influenced, based on several factors. Thus, modelling and testing these factors might lead to better understanding of (how & when) to use particular tool in software development environment.

Accordingly, and based on the witnessed high uncertainty and complexity in modern software development processes, Media Richness Theory will be adopted as a theoretical framework. We compared the applicability of MRT throughout both Agile and Waterfall software development stages and the influence of some key factors on the tool selection as shown in **Figure 1**.
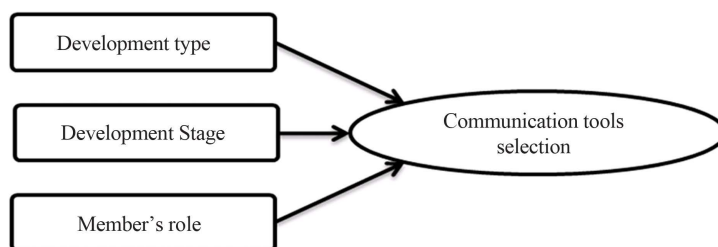

Figure 1. Proposed study model.

In our model as shown in **Figure 1**, we assume that communication tools selectivity can be influenced by three factors as following:

1) Development type: which means that software development teams communicate differently based on the development approach that they adopt, for example, communication channels in Agile approach is different from Waterfall approach in terms of the means that have been used in the communication process and information transfer.

2) Development stage: which means that software development teams communicate differently based on the development stage as we rely on certain tools and channels in certain stages and we change these tools once we moved to the subsequent stage.

3) Member's role: which means that software development members are different from each other in terms of their preferences in selecting certain tools based on their roles in the project, for example: developers might prefer emails over phone calls, whereas analyst might prefer face to face over emails during the project's lifecycle.

4) Media Richness level of certain communication tool might be influenced by the changes that might happen to the communication selectivity in software development environments.

Accordingly, our research questions are:

Q1. What is the influence of the project type (Agile or Waterfall) on the selection of communication tools during the development stages?

Q2. What is the influence of a project member's role on the selection of communication tools during the development stages?

Q3. How do project stages influence the selection of communication tools?

Q4. How do Media Richness rankings differ depending on the communication direction?

Q5. What is the updated rank ordering of communication tools based on Media Richness theory criteria?

4. Theoretical Framework

Generally, organizations are human interaction systems. Information is conveyed through symbols and language systems that are used to interpret situations and adjust behaviour [22]. Communication can be a big challenge within software development settings [23]; nevertheless, people's personalities, culture and attitudes can also increase this challenge's scale throughout software development stages [1]. In software development, many communication tools can be used in order to exchange project information and requirements. Furthermore, Media Choice Theory has argued for the use of certain tools over others for certain activities [24]. A number of tools can be used for particular tasks whereas others cannot [25]. Coordinating teams' dependencies within software development work can result in coherent software [7], while sharing information and knowledge would pave the way for proper coordination and communication [26] [27].

Media Richness Theory (MRT) explains the way that we can reduce uncertainty and ambiguity within the organisational context through proper use of the communication medium. MRT helps in evaluating the richness of certain communication mediums in conveying messages between people; for example, face-to-face is richer than phone conversation. Accordingly, Cockburn in 2002 used MRT to develop a communication model for use in Agile practices [28].

Moreover, this theory provided an insight into the correlation between the message and the medium and how homogeneity between them can lead to a successful communication process. In addition, this theory consists of four properties: the medium's capability for immediate feedback, the number of cues and channels available, the language variety, and the degree to which intent is focused on the recipient [19]. Based on these four factors, the richness level can be evaluated and rated [24].

The richness level of each tool can be measured by how fast any ambiguity has been clarified and the issue understood; if that happens relatively quickly, it means the medium is considered richer, and vice versa [19]. Efficient communication occurs when task properties and media richness matches [24]. Furthermore, media richness theory suggests that tasks with high ambiguity and uncertainty should be treated by richer media, while unequivocal and certain tasks can be accomplished better with a leaner medium [19]. Although Dennis, Valacich, Speier, & Morris (1991) found that Media Richness Theory was built on the presumption that increased richness is linked to increased social or physical presence, Media Richness Theory can still provide a proper correlation between the effectiveness of the communication tools and the desired situation [29].

5. Communication in the Waterfall Model

The Waterfall approach was developed by Royce (1970) to ensure success in software project development and is used widely in Software Engineering. It consists of a set of sequential development stages (like a Waterfall). These stages are: System Requirement Analysis, Design, Implementation (developing the code that constitutes the software), Testing and debugging, Deployment (instillation) and Maintenance as shown in **Figure 2**.

   Project members are required to complete each stage before the project can progress to the next stage. This model adopts a linear approach where the outcome of each stage acts as the input for the next stage sequentially [31]. More precisely, communication in Waterfall development occurs based on the purpose of each stage; for example, communication happens extensively in the System Requirement Analysis stage in order to clarify the users' requirements and to manage their expectations [32]. Many communication tools have been proven to be useful in the software development business such as face to face, instant messaging, phone call, and video conference [1] [9] [15] [33]. However, describing the best use of these tools and explaining the proper place to facilitate them has not been given much consideration in Waterfall software development studies.

6. Communication in Agile Model

Instead of the sequential stages followed in the Waterfall model, Agile adopts an incremental approach to build software gradually and dynamically by conducting multiple incremental iterations [34]. Practically, The Agile approach adopts the stages of software development but in an iterative manner, and an Agile team can produce a working release from the first development stage as shown in **Figure 3**.

   Teamwork and project management principles are employed inherently in Agile approaches [35]. The Agile manifesto emphasises easy, direct and constant verbal communication, for example "Individuals and interactions over processes and tools", in all software development iterations [36]. Based on the Agile manifesto principles, direct and immediate communication channels are highly preferred during Agile iterations [37] [38]. The Agile approach emphasizes people, communication, collaboration and the willingness to accept all users' requirements since project stake holders work iteratively on a daily basis [38].
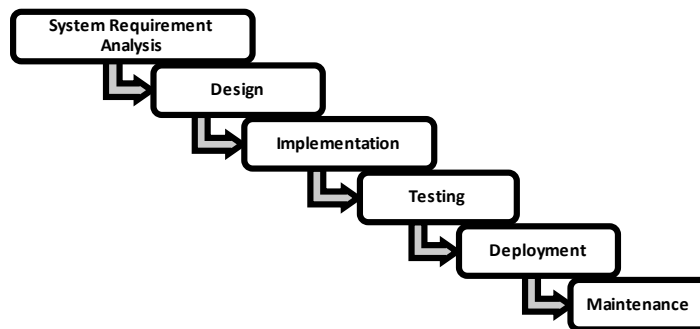

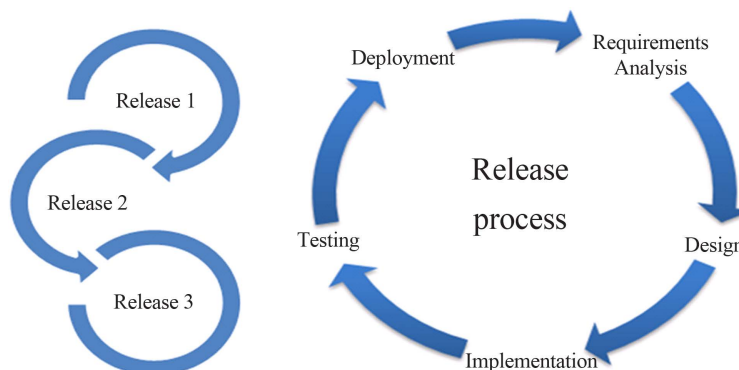**Figure 2.** Waterfall model (Royce, 1970) [30].

Figure 3. Agile model [37].

Communication during software development stages has been given a moderate consideration in literatures; Cockburn in 2002 modelled the effectiveness of communication tools throughout project development (**Figure 4**). The Agile processes provide regular and repeated interaction activities for project members in either co-located or distributed software settings, this, can help in moving information between people easily and steadily, and hence requirements can be thoroughly validated and uncertainty level can be minimised. Furthermore, working together on a daily basis can motivate project members to share their mutual responsibilities and commitment towards the organisation [28].

Amber (2002) suggests some factors that might influence the quality of communication within Agile development which are:

1) Physical proximity: the physical distance will be inversely proportional to the communication quality and clarity.

2) Temporal proximity: how people from different time zones can communicate at a preferable and convenient time.

3) Amicability: the willingness to communicate and speak to each other.

4) Tool applicability and accessibility

5) Anxiety and personal preferences [39] as shown in **Figure 5**.

Although there is some general consensus regarding the tools to be used in Agile iterations, these tools should be examined and compared in terms of their applicability indifferent circumstances.

7. Communication in Agile and Waterfall

Although project behaviour is significantly different between Agile and Waterfall, both approaches aim to deliver the targeted software on time within the agreed cost and quality levels. Agile approaches allow the client to make changes, and developers to add features that will keep the product up to date (it is well known for being dynamic and allowing user engagement); however, Waterfall can be the sole available option for some organisa-
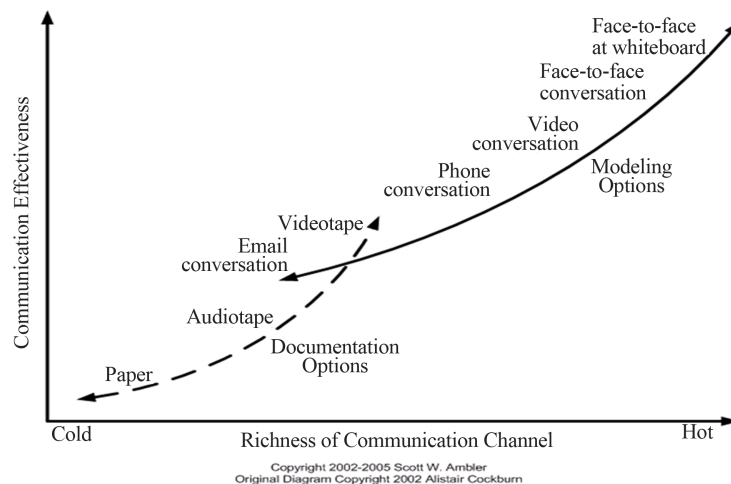

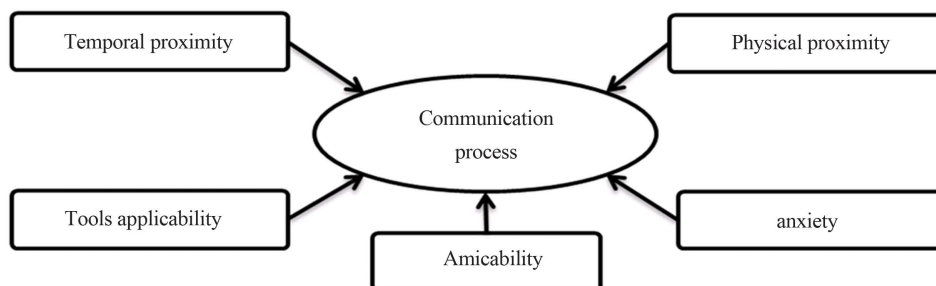**Figure 4.** Cockburn MRT ranking [28].

Figure 5. Communication influential factors model.

tions due to organisational considerations.

In terms of communication, Verner & Babar state that the two-way and immediate communication in the Agile approach makes it faster than the Waterfall development approach [40]. In addition, the User Centered Design (UCD) approach has adopted several communication techniques as it bridges the gap between developers and users to ensure an easier time in discussion and in providing examples. Agile communication modes have allowed the software team to narrow the gap between uncovering usability issues and acting on those issues by incorporating changes into the product [11]. Another recent comparison was conducted by Petersen & Wohlin revealing many advantages as a result of moving from plan-driven methods to Agile; some of these advantages concerned communication effectiveness [41].

Although a number of studies have prioritized some development approaches over others in terms of the communication effectiveness criterion, there is an increased need to give deep explanation about how to facilitate communication tools effectively and how to pave the way for communication parties to communicate properly in different situations.

In this study, we will try to evaluate the use of communication tools through Agile and Waterfall methods by adopting MRT principles and criteria. The influence of project type, project stage and member role will be measured against the selection of communication tools.

8. Methodology

This study was introduced to re-evaluate the applicability of Media Richness Theory, and how some recent factors can affect this theory's applicability in both Agile and Waterfall approaches.

Media Richness Theory concerns the effectiveness of communication tools as a medium between people and therefore this theory is affected by the fitness of the media and the characteristics of the communication task [19]. MRT's applicability can be measured by evaluating four main criteria: 1) immediate feedback; 2) capacity to transmit multiple cues; 3) language variety; 4) personal focus [22].

In order to increase the validity of this study's results, a mixed methodology approach (qualitative and quantitative) was adopted. The *qualitative* part was used to highlight the most recent communication tools being used in the software development sector, as well as some influential factors on software development team communication.

The quantitative part was designed as a web-based multilingual survey, questions were translated into two Languages (Arabic & English) in order to increase participants' understanding of our questions, and hence, to increase the validity. The survey was distributed among software developers worldwide. Following the methodological steps of (Daft, Lengel, & Trevino, 1987), the survey was designed based on actual development situations through a communication incident. This incident consisted of many stages similar to those in software development; in each stage the participant was put in a situation, and they were expected to use one communication tool in order to respond to that situation.

Furthermore, Cockburn's model as shown in **Figure 4** was re-evaluated by adding more recent communication tools (instant messaging, paired face to face meeting and recent video/audio conference technologies), and then finding out the best place to locate them in Cockburn's model; nevertheless, the existing order of communication tools in Cockburn's model will be examined against the new inserted communication tools' order, as Cockburn's study was conducted in 2002 and is in need of updating.

9. Survey Design

A web-based survey was designed based on three themes of questions:

• Part 1: Communication during stages: based on every stage in Information Systems development (IS), each participant must choose one communication tool that can help in tackling each stage.

• Part 2: Ranking communication tools based on the participant's role in a communication incident whether he/she is communicating with a manager, developer or client.

• Part 3: Ranking communication tools based on the criteria in Media Richness Theory which are feedback, multiple cue transmission, language variety, and personal focus.

The tool ranking score was calculated based on the preference for each tool from the perspective of the participants; for instance, if the tool chosen with the highest preference, to be ranked as number 1 in the order, then it will be scored as 9 based on this scoring formula (n + 1) – x, where n is the number of displayed tools count; x

is the selected order. The final score for each communication tool will be calculated as shown in Formula (1)

$$\sum (n+1) - x \qquad (1)$$

10. Results

10.1. Interviews Results

We invited around 20 project managers to participate in interviews of whom 13 accepted our invitations. Online, phone and email conversations were used to get the answers to four open-ended questions, these four questions attempted to highlight the new communication tools used in the market, and whether these tools were applicable or not throughout project stages by all team members. All of the project managers were male as we didn't obtain acceptance from any invited females. According to the working sector, 38.4% (5 interviewees) of the sample are working for the public sector, and 62% (8 interviewees) of them are working for the private sector.

All interviewees agreed upon the importance of using the fast and immediate communication tools as a protective shield against any threatening ambiguity. Moreover, there is a consensus that the bureaucratic process in getting information from the end user has significantly impeded project's progress.

In terms of the communication tools that are being used currently in their project, the most obvious theme indicates that the recent development of Information and Communication Technologies (ICT) tools has given enormous advantages to the communication processes within software development stages such as instant messaging and live chatting (Skype, Google Talk, …etc.). Furthermore, some mobile applications such as Whatsapp have been used informally in IS development because of security and privacy concerns. However, most of this study's interviewees do not like using public communication applications (Whatsapp for example) for business as everything must be documented and coordinated properly.

Interestingly, about 46% of the interviewees have recommended using applications designed specifically for software development communication processes such as: Fogbugz, Bug Genie and bugzilla. These applications are concerned with 1) Reporting bugs directly; 2) Tracking changes without contacting developers personally. However, the other interviewees have different opinions about using this tracking software due to the client commitment to use them continuously. Furthermore, 61.5% of the interviewees agreed upon splitting face-to-face meetings into three types (formal, informal, and paired meeting) as every type can be "serviceable" in certain situations.

Furthermore, interviewees were asked about their preferred way to be communicated during project development, whether they prefer particular tools (email for example) over all other tools. The results show that they are relatively different from each other. In addition, some of them conditioned the use of a particular tool with one of the following factors (personal characteristic, subject priority or project situation).

All the interviewees were inducted initially by asking them "How would you prefer to communicate during the IS development process?", and as we expected, an obvious theme emerged consisting of undecided terms with more options based on several influential factors.

As expected, face-to-face communication was the dominant tool for most of the study's interviewees; however, this type of communication is conditioned by the subject importance, severity and equivocality levels.

Finally, the most mentioned communication tools used in IS development are: face-to-face with dashboard, face-to-face formal, face-to-face paired, webinar, mobile, phones, email, texting, instant messaging, and letters. Email has been recorded as the most common tool among all our interviewees.

The qualitative data has provided a clear way for the quantitative data methodology to be conducted in order to answer this study's questions. These questions concern the influence of Project type, member role and project stage on the selection of communication tools and hence the richness level of these tools from a Media Richness Theory perspective.

10.2. Quantitative Results

In-depth interviews with 13 project managers have been conducted prior to our web-based questionnaire; interviews have contributed to our survey design in terms of the current applicable communication tools and practices. Thereafter, feedback from 66 participants (worldwide software developers) was gathered and analysed accordingly. 20 questions were posted in one questionnaire which was made available to the participants online for 4 weeks.

The questionnaire was designed in three parts:
1) Communication during development stages.
2) Communication tool ranking based on the type of project stakeholders.
3) MRT-based communication tool ranking.

As shown in **Table 1**, over two-thirds of the participants fall into two main roles in software development (Project manager & Developer/designer/networking) with a percentage of 37.9 and 40.6 respectively. Furthermore, in terms of their age group, as expected, nearly two-thirds of those surveyed were between 35 and 54 years old, followed by participants whose ages were between 25 and 34.

Since the survey's factors are categorical such as project type, member role and communication tools, the Chi-square (χ^2) criterion was originally chosen to test this study's hypothesis; however, due to the small count number in some selected communication tools of "less than five", we used Fisher's Exact Test based on Monte Carlo simulation [42]. Significant influence occurs if Probability (P-value) is less than 0.05, where a 99% Confidence Interval (CI) can be calculated.

11. Analysis and Discussion

1) What is the influence of the project type (Agile or Waterfall) on the selection of communication tools during the development stages?

A Fisher's Exact Test for association was conducted between project type and the selection for specific communication tools overall project stages. There was a statistically significant association between those two variables with P values less than 1.00×10^{-6} as shown in **Table 2**. More investigation has revealed that the differences occur mostly in using (paired meeting) and (video-conference) tools.

Since Agile methods rely mostly on repeated iterations during all project versions or "sprints", the difference from the Waterfall approach in selecting the proper communication tools can be attributed to the overarching need to adopt Agile's four principles (more meetings, fewer documents, speed and collaborative progress) [12].

Table 1. Questionnaire participants role and age groups.

Role	Frequency	Percent
Project role		
Project manager	25	37.9
Business/logic analyst	5	7.6
Developer/designer/networking	27	40.9
Database administrator	5	7.6
Coordinator	4	6.1
Total	66	100.0
Age group		
18 - 24	2	3.0
25 - 34	18	27.3
35 - 54	45	68.2
>55	1	1.5
Total	66	100.0

Table 2. The influence of project type on communication tools selection.

Fisher's exact test according to stages	Monte Carlo significance (2-sided)		
	P	CI 99% lower	CI 99% upper
Project type over stages	$<1.00 \times 10^{-6}$	$<1.00 \times 10^{-6}$	4.60E−04

Moreover, this significance level in our analysis can be seen as a positive direction to changing people's behaviour throughout development stages in Agile projects.

2) What is the influence of a project member's role on the selection of communication tools during the development stages?

A. Member's role on tools selection:

A Fisher's Exact Test for association was conducted between members' roles in the project and their selection for specific communication tools overall project stages. Interestingly, significance level has been noticed only in the implementation phase at the level of $P = 0.0085$. Practically, the implementation stage is seen as a crucial stage. This is based on our finding that people differ from each other in this stage depending on their role in the project; this significance level might be attributed to their preferring to complete their tasks on time.

B. Combined variables:

Regression analysis shows that combining multiple independent variables can lead to some unexpected results. Observationally, *project type*, *member's role and stage*, taken together can explain some of our sample's selection of communication tools, since $R^2P = 4.89 \times 10^{-12}$ where $R^2 = 0.09$.

The development team's selection from a range of communication tools appears, statistically, to be affected by project type, member's role and stage together; this influence could lead to a better understanding of the dynamics of communication tools' selection.

3) How do project stages influence the selection of communication tools?

Our statistical analysis shows that use of "Informal face to face" in the first stage (inception meeting) is at least twice as frequent as in other project's stages. As well as, more than three times more common than all the stages together. The utilization of this tool is very important in both Agile and Waterfall as the development team are meeting with presentation tools (whiteboards, projectors, etc.) in order to describe the project's scope and main requirements; the project's progression plan and stages would be drawn accordingly.

4) How do Media Richness rankings differ depending on the communication direction?

Participants were asked to rank all communication tools based on the type of person that they are communicating with (client, colleague or their manager). A scaling process has been adopted to find out the weight of each communication tool against the other tools while they are being used by the same person. The results are presented in **Figure 6** in a mixed radar chart to show two main points (Convergence point, and Divergence point). Thus, we could compare the usage of one tool with different people:

a) Convergence points, where all lines come to specific points. Managers, clients and colleagues are communicating equally with some communication tools; for example: face-to-face formal, videotaping, email, mobiles and letters. Generally, it is important to note that these tools are different from each other in their importance.
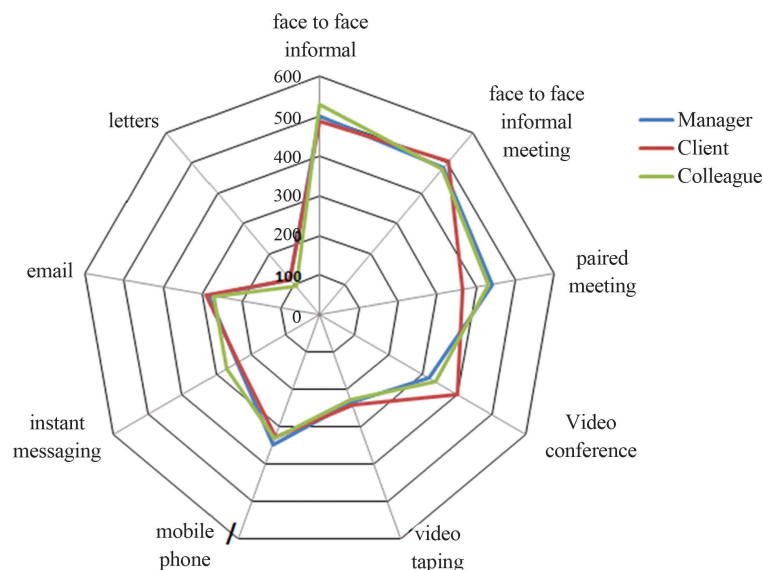

Figure 6. Weight of each communication tool against the other tools per stakeholder.

b) Divergence points: from the chart we can see that communication with clients has less stability compared to communication with either managers or colleagues. Paired meeting with clients is preferred less, and conversely, using video conferencing like Skype and video chatting is preferred to communication with managers and colleagues.

Noticeably, using instant messaging between team members has been observed to be more preferred to using it with managers and clients. Moreover, using informal meetings has recorded a higher preference between team members than communicating with manager and clients.

5) What is the updated ranking order for communication tools based on Media Richness theory criteria?

In order to answer this question we had to follow two steps. Firstly, re-evaluate communication tools based on MRT criteria (feedback, cues and signs, language clarity and feeling & emotion transfer) [19] [43]. Secondly, rank communication tools based on the average of all MRT criteria.

A. Re-evaluating MRT effectiveness in Software development setting:

Every criterion from the MRT evaluation process has been scaled and illustrated separately in a bar chart; this scaling process is not related to any group or independent variable, and it covers the entire sample's evaluation as shown in the **Figures 7-10**.

Regarding the effectiveness of feedback, mobile/phones are obviously becoming competitive with the three face-to-face types and video conferencing tools. Predictably, formal letters are the least favoured for providing immediate feedback, as shown in **Figure 7**.

Regarding the effectiveness of sending cues and signs in order to easily transfer information, informal face-to-face is on the top of our sample's preferences, as providing the best way of sending cues and explaining more information to the other party. Email and formal letters are the least preferred tools, as shown in **Figure 8**.

Moreover, tools are ranked in descending order based on the effectiveness of language clarity; ranking has taken relatively the same order as the previous criterion (the effectiveness of sending cues and signs), as shown in **Figure 9**.

Lastly, in terms of the effectiveness of transferring feeling and emotions, visualisation tools are competing in the first four preferred tools, while conversely; non-visualising tools have low scorings, as shown in **Figure 10**.
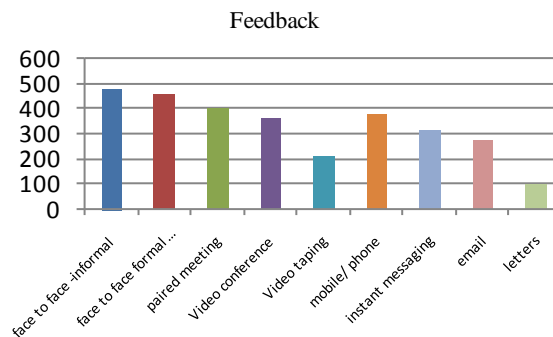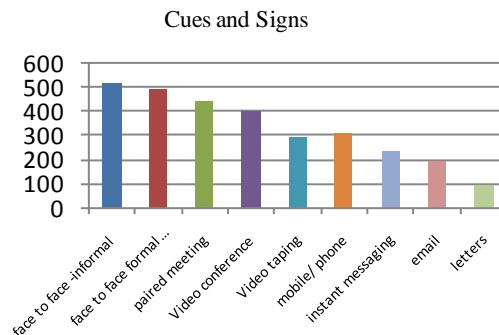

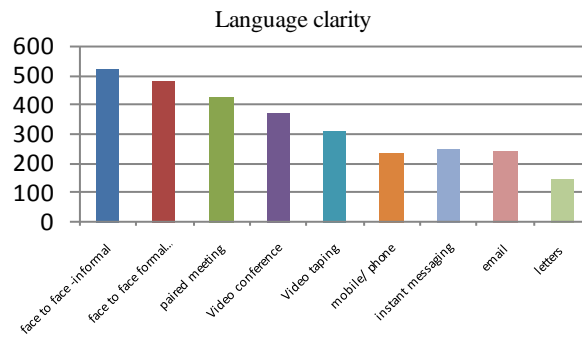**Figure 7.** Feedback-based ranking.

Figure 8. Cues-based ranking.

Language clarity

Figure 9. Language clarity-based ranking.
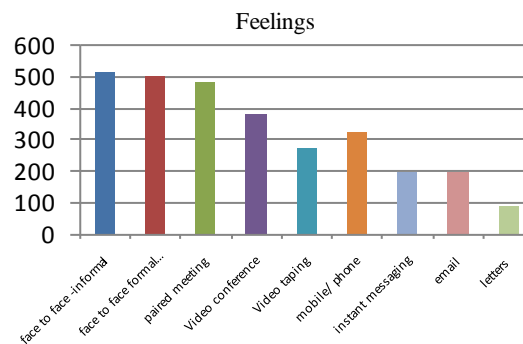
Feelings

Figure 10. Feeling transfer-based ranking.

More importantly, all communication tools follow roughly the same order throughout the evaluation's four criteria; informal face-to-face is top of the list, followed by the formal face-to-face meeting. Predictably, formal letters was the least favourable option chosen amongst our sample in terms of the ability to meet all four evaluation criteria.

Interestingly, interaction between the four criteria has revealed a clearer explanation for all communication tools, as shown in **Figure 11**.

The above Radar chart can be described based on two obvious points:

1) Convergence points, where all lines come to a specific point, which means that the assigned communication tool has equally the same level of preference in the four criteria. For example: informal face to face has almost the same level in the four dimensions.

2) Divergence points, where all lines are distributed over different degrees, which means that the assigned tool has different levels of severity over the four MRT dimensions. For example: the Feedback dimension has moved unstably between communication tools, and noticeably it can be seen as the best for mobile/phone tools and conversely seen at the bottom for the other tools. Furthermore, instant messaging and paired meeting have recorded significant divergence levels between MRT's four criteria. Expectedly, the formal letters tool has recorded a very low preference level throughout the four dimensions.

B. Updated Ranking for use of communication tools based on MRT perception

Using communication tools in the software development setting is affected by several factors as shown in the statistical section (Quantitative results); however, these tools can be ranked based on the overall average of the MRT's four dimensions (feedback, cues and signs, language clarity and feeling & emotion transfer) as shown in **Figure 12**.

The above bar chart shows that face-to-face tools (informal, formal and then paired) are the most preferable communication tools respectively. Moreover, letters was the least preferred tool. More precisely, our updated ranking order for communication tools is shown in **Figure 13**.

If we compare between our new ranking in **Figure 13**, with the previous ranking introduced in **Figure 4** by Cockburn in 2002 [28], we can see that our new ranking order is slightly different from the Cockburn's ranking; as some tools like audio taping have been discarded, and replaced by instant messaging tools. In addition, the
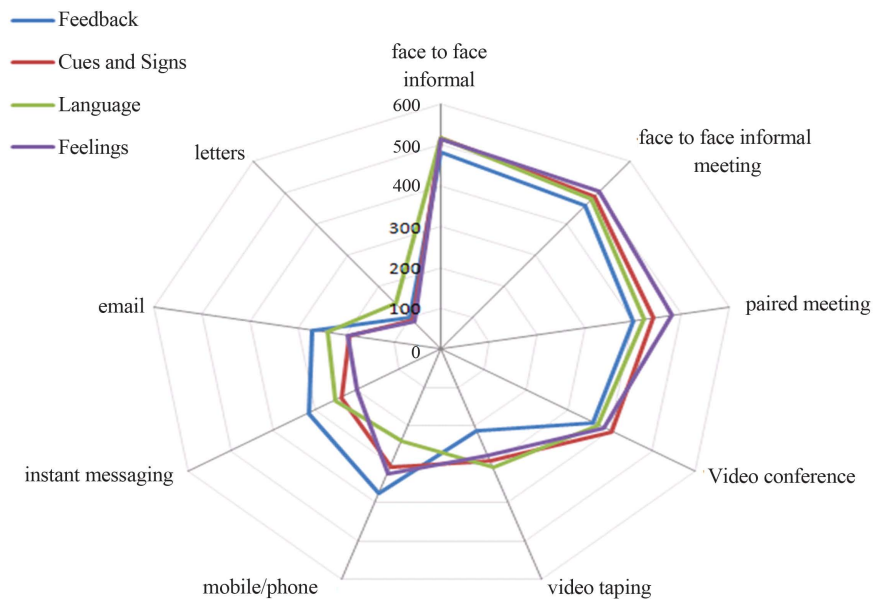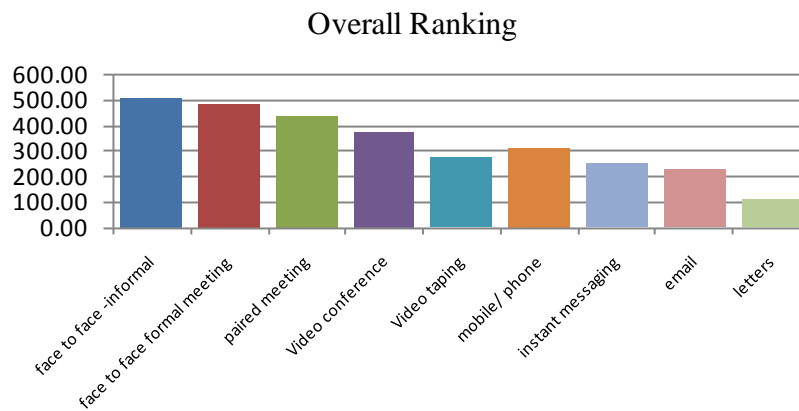
Figure 11. Communication tools interactions.

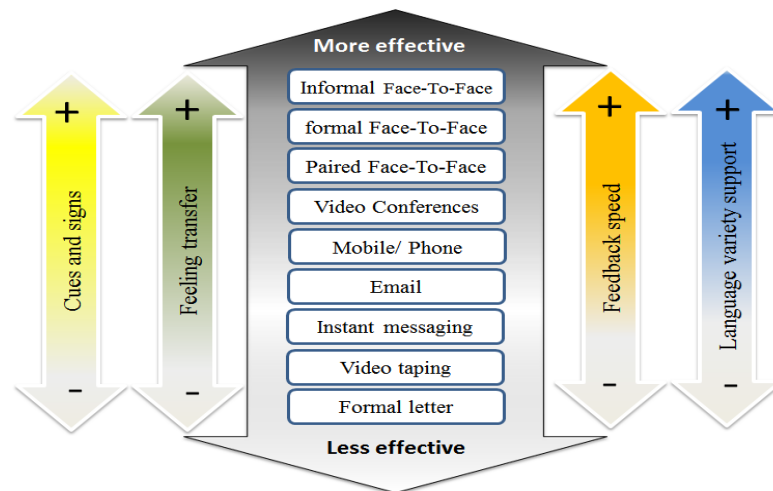**Figure 12.** MRT overall ranking.

Figure 13. Updated ranking order for communication tools.

videotaping tool has scored very poorly based on the MRT criteria but still shows a better score than the formal letters tool. Expectedly, email use has recorded a higher level in our study than Cockburn's study.

12. Conclusions

The aim of this study is to re-evaluate Media Richness Theory, Cockburn (2002), in the software development setting; the influence of some factors like project type (Agile & waterfall), project stage and members' role were tested statistically and the relationships amongst them were discussed in depth. Ranking communication tools based on the other party's role (manager, client or colleague) has revealed some interesting results about the best tool for each pair of roles. In addition, evaluating all communication tools based on MRT criteria has given a better understanding of how effective these tools are in software development environment.

We conclude that using the best communication is affected by several factors. Thus we cannot rely on one tool in all project situations. These differences in using communication tools could be attributed to the type of transferable information or even to personal preferences.

Further research could lead to more interesting results if communication tool preferences are linked to cultural and geographical factors.

References

[1] Atwood, M.E., Burns, B., Gairing, D., Girgensohn, A., Lee, A., Turner, T., Alteras-webb, S., Zimmermann, B. and Plains, W. (1995) Facilitating Communication in Software Development. *Proceedings of the* 1*st conference on Designing Interactive Systems: Processes*, *Practices*, *Methods*, *& Techniques*, 65-73.

[2] Lagerberg, L., Skude, T., Emanuelsson, P., Sandahl, K. and Stahl, D. (2013) The Impact of Agile Principles and Practices on Large-Scale Software Development Projects: A Multiple-Case Study of Two Projects at Ericsson. *Empirical Software Engineering and Measurement*, ACM/IEEE International Symposium, 348-356.

[3] Linberg, K.R. (1999) Software Developer Perceptions about Software Project Failure: A Case Study. *Journal of Systems and Software*, **49**, 177-192. http://dx.doi.org/10.1016/S0164-1212(99)00094-1

[4] Gopal, A., Mukhopadhyay, T. and Krishnan, M.S. (2002) The Role of Software Processes and Communication in Offshore Software Development. *Communications of the ACM*, **45**, 193. http://dx.doi.org/10.1145/505248.506008

[5] PMBOK_GUIDE (2004) Project Management Body of Knowledge. 3rd Edition, Newtown Square, Pennsylvania 19073-3299 USA: The Project Management Institute.

[6] Shannon, C. (1948) The Mathematical Theory of Communication. *Bell System Technical Journal*, **27**, 379-423, 623-656.

[7] Kraut, R.E. and Streeter, L.A. (1995) Coordination in Software Development. *Communications of the ACM*, **38**, 69-81. http://dx.doi.org/10.1145/203330.203345

[8] Pleshkevich, E.A. (2010) Information and Communication Activity, Its Essence, Structure, and Types. *Automatic Documentation and Mathematical Linguistics*, **44**, 246-248. http://dx.doi.org/10.3103/S000510551005002X

[9] Pazos, P., Chung, J.M. and Micari, M. (2012) Instant Messaging as a Task-Support Tool in Information Technology Organizations. *Journal of Business Communication*, **50**, 68-86. http://dx.doi.org/10.1177/0021943612465181

[10] Sankaran, S. and Agarwal, D.R. (2012) Rethinking Project Management Goals and Methods to Suit Service. *Proceedings of the* 56*th Annual Meeting of the ISSS*.

[11] Sy, D. (2007) Adapting Usability Investigations for Agile User-Centered Design. *Journal of Usability Studies*, **2**, 112-132.

[12] Pathak, K. and Saha, A. (2013) Review of Agile Software Development Methodologies. *International Journal of Advanced Research in Computer Science and Software Engineering*, **3**, 270-276.

[13] Blomkvist, S. (2005) Towards a Model for Bridging Agile Development and User-Centered Design. In: *Human-Centered Software Engineering—Integrating Usability in the Software Development Lifecycle*, Springer, Uppsala, Sweden 219-244. http://dx.doi.org/10.1007/1-4020-4113-6_12

[14] Fox, D., Sillito, J. and Maurer, F. (2008) Agile Methods and User-Centered Design: How These Two Methodologies Are Being Successfully Integrated in Industry. *Agile Conference*, Toronto, 4-8 August 2008, 63-72. http://dx.doi.org/10.1109/Agile.2008.78

[15] Bolici, F., Howison, J. and Crowston, K. (2009) Coordination without Discussion ? Socio-Technical Congruence and Stigmergy in Free and Open Source Software Projects. *Socio-Technical Congruence Workshop in conj Intl Conf on Software Engineering*, Vancouver, Canada.

[16] Franca, A.C.C., Carneiro, D.E.S. and Da Silva, F.Q.B. (2012) Towards an Explanatory Theory of Motivation in Software Engineering: A Qualitative Case Study of a Small Software Company. 26*th Brazilian Symposium on Software Engineering*, Natal, 61-70. http://dx.doi.org/doi:10.1109/SBES.2012.28

[17] Cataldo, M., Herbsleb, J.D. and Carley, K.M. (2008) Socio-Technical Congruence: A Framework for Assessing the Impact of Technical and Work Dependencies on Software Development Productivity. *Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, ACM, Kaiserslautern, 2-11. http://dx.doi.org/10.1145/1414004.1414008

[18] Valacich, J.S., Mennecke, B.E., Wachter, R.M. and Wheeler, B.C. (1994) Extensions to Media Richness Theory: A Test of the Task-Media Fit Hypothesis. *Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences*, **4**, 11-20.

[19] Daft, R.L. and Lengel, R.H. (1986) Organizational Information Requirements, Media Richness and Structural Design. *Management Science*, **32**, 554-571. http://dx.doi.org/10.1287/mnsc.32.5.554

[20] Short, J., Williams, E. and Christie, B. (1976) The Social Psychology of Telecommunications. John Wiley and Sons Ltd., Hoboken.

[21] Dennis, A.R. and Valacich, J.S. (1999) Rethinking Media Richness: Towards a Theory of Media Synchronicity. *HICSS*-32. *Proceedings of the 32nd Annual Hawaii International Conference on System Sciences*, **1**, 1017.

[22] Daft, R.L., Lengel, R.H. and Trevino, L.K. (1987) Message Equivocality, Media Selection, and Manager Performance: Implications for Information Systems. *MIS Quarterly*, **11**, 355-366.

[23] Allen, T.J. (1977) Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization. MIT Press, Cambridge.

[24] Niinimaki, T., Piri, A., Lassenius, C. and Paasivaara, M. (2010) Reflecting the Choice and Usage of Communication Tools in GSD Projects with Media Synchronicity Theory. 5*th IEEE International Conference on Global Software Engineering* (*ICGSE*), IEEE Computer Society, Princeton, 3-12. http://dx.doi.org/10.1109/ICGSE.2010.11

[25] Gutwin, C., Penner, R. and Schneider, K. (2004) Group Awareness in Distributed Software Development. *Proceedings of the* 2004 *ACM Conference on Computer Supported Cooperative Work*, ACM, Chicago, 72-81. http://dx.doi.org/10.1145/1031607.1031621

[26] Damian, D., Helms, R., Kwan, I., Marczak, S. and Koelewijn, B. (2013) The Role of Domain Knowledge and Cross-Functional Communication in Socio-Technical Coordination. 35*th International Conference on Software Engineering*, IEEE, San Francisco, 442-451.

[27] Dourish, P. and Bellotti, V. (1992) Awareness and Coordination in Shared Workspaces. *Proceedings of the* 1992 *ACM Conference on Computer-Supported Cooperative Work*, ACM, Toronto, 107-114. http://dx.doi.org/10.1145/143457.143468

[28] Cockburn, A. (2002) Communicating, Cooperating Teams. *Agile Software Development*, Vol. 2006, Addison-Wesley Longman Publishing Co., Inc., Boston, 90-100.

[29] Dennis, A.R., Valacich, J.S., Speier, C. and Morris, M.G. (1998) Beyond Media Richness: An Empirical Test of Media Synchronicity Theory. *HICSS* '98: *Proceedings of the Thirty-First Annual Hawaii International Conference on System Sciences*, **1**, 48-57. http://dx.doi.org/10.1109/HICSS.1998.653082

[30] Royce, W.W. (1970) Managing the Development of Large Software Systems: Concepts and Techniques. *Proceedings of the* 9*th International Conference on Software Engineering*, **26**, 328-338.

[31] Atkinson, S. and Benefield, G. (2013) Software Development: Why the Traditional Contract Model Is Not Fit for Purpose. In: Ralph, J. and Sprague ,H., Ed.s, 46*th Hawaii International Conference on System Sciences*, IEEE Computer Society, Wailea, Maui, Hawaii, 1-10.

[32] Caltrans (2007) Project Ommunication Handbook. 2nd Edition, Office of Project Management Process Improvement, Californai.

[33] Page, X., Knijnenburg, B.P. and Kobsa, A. (2013) FYI: Communication Style Preferences Underlie Differences in Location-Sharing Adoption and Usage. *Proceedings of the* 2013 *ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ACM, Zurich, 153-162. http://dx.doi.org/10.1145/2493432.2493487

[34] Shore, J. and Warden, S. (2008) The Art of Agile Development. 2nd Edition, O'Reilly Media, Inc., Sebastopol.

[35] Cockburn, A. and Highsmith, J. (2001) Agile Software Development: The People Factor. *Computer*, **34**, 131-133.

[36] Beck, K., *et al*. (2001) The Agile Manifesto. *Agile Alliance*. http://agilemanifesto.org/

[37] Larman, C. (2003) Agile and Iterative Development: A Manager's Guid. In: Cockburn, A. and Highsmith, J., Eds., 2nd Edition, Pearson Education, Boston, 25-28.

[38] Agile_Manifesto (2001) Manifesto for Agile Software Development. http://www.agilemanifesto.org/

[39] Ambler, S. (2002) Agile Modeling: Effective Practices for Extreme Programming and the Unified Process. John Wiley & Sons, Hoboken.

[40] Huo, M., Verner, J., Zhu, L. and Babar, M. A. (2004) Software Quality and Agile Methods. *Proceedings of the* 28*th Annual International Computer Software and Applications Conference*, 2004. *COMPSAC* 2004, IEEE Computer Society, Hong Kong, 520-525. http://dx.doi.org/10.1109/CMPSAC.2004.1342785

[41] Petersen, K. and Wohlin, C. (2010) The Effect of Moving from a Plan-Driven to an Incremental Software Development Approach with Agile Practices. *Empirical Software Engineering*, **15**, 654-693. http://dx.doi.org/10.1007/s10664-010-9136-6

[42] Fisher, R.A. (1922) On the Interpretation of χ2 from Contingency Tables, and the Calculation of P. *Journal of the Royal Statistical Society*, **85**, 87-94. http://dx.doi.org/10.2307/2340521

[43] Dennis, A.R. and Kinney, S.T. (1998) Testing Media Richness Theory in the New Media: The Effects of Cues, Feedback, and Task Equivocality. *Information Systems Research*, **9**, 256-274. http://dx.doi.org/10.1287/isre.9.3.256

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or Online Submission Portal.