



Prioritizing Usability Data Collection Methods

Teena Tiwari Parmar¹, Dr. Kshama Paithankar²

¹Assistant Professor, ²Professor,

Shri Vaishnav Institute of Management, Indore, Madhya Pradesh, INDIA

Abstract: Various data collection methods are available to serve the purpose of fulfilling usability requirements. Though all these methods are equally useful for data collection, the scale of influence of these methods varies in different classes of software in order to fulfil usability requirements. Observations strengthen the fact that to attain software usability, usability requirements must be determined and fulfilled throughout software development. Therefore, it is required to identify the Usability Data Collection Methods (UDCM) with preferences. In this paper, we propose to prioritize aforesaid UDCMs by computing some metrics referring different classes of software. Case study has also been performed considering three classes of software. The proposed work is useful to understand the importance of UDCMs and further for usability measurement considering specific nature of software. Also, it will lead to provide ease to developers and thereby maximizing user satisfaction as well.

Keywords: Potential Data Collection Methods (PDCM), Applicability Factor, Productivity Factor, Total Productivity Factor, Usability Data Collection Methods (UDCM).

I. Introduction

Usability requirements capture the characteristics to build a usable system from user perspective. Understanding usability requirements is an integral part of usable software development [1]. It is widely observed that successful systems and products begin with identifying and fulfilling usability requirements [2]. Therefore, a strong need has emerged to identify the Data Collection Methods (DCM) to satisfy phase wise usability requirements. It will provide benefits such as; increased productivity, enhanced quality of work, reductions in support and training costs, and finally improved user satisfaction ([3], [4]). Also, incorporating usability requirements in the development process can reduce the risk of failure [5]. Further, fulfilling these usability requirements data must be collected for which there exist various DCM. It has been observed that usability requirements vary phase wise in the development process and the kind of final product i.e. to be usable software ([6], [7]). This paper is intended to find the Potential Data Collection Methods (PDCM) that are defined as UDCM for fulfillment of usability requirements ([8], [9]). Mainly, prioritization of UDCMs is proposed in this paper. Section-II covers the identification of PDCM towards prioritization of UDCM. Section-III deals with the prerequisites for prioritization of UDCM. In section-IV some useful metrics are proposed with computation along with the steps for UDCM prioritization. Case study illustrating the proposal quantitatively is presented in section-V. Finally, we end up with conclusion in section-VI.

II. Identification of PDCM

It has been observed that various DCM are available for collecting data related to usable software development ([8], [9]). However, all of these methods are not useful for fulfilling usability requirements completely. Therefore, to identify PDCM amongst all DCM the scale of influence is used considering Mandatory (Man), Not Required (NR) and Optional (Opt) influence for a specific software class. It is suggested that a method is PDCM if its influence in all phases of software development is either mandatory or optional [9]. Further, priority is assigned to the phases of software development. Considering analysis as the most important phase of software development, this phase has assigned the highest priority. Similarly, according to the importance of the phase, priorities are assigned. Finally, collaborating scale of influence and phase wise priority, PDCM are identified for each class of software and termed as UDCM as shown in Table-1.

Table-1: Usability Data Collection Methods (UDCM)

Potential DCM for Aps	Analytical Preference of Methods for Aps	Potential DCM for UIn	Analytical Preference of Methods for UIn	Potential DCM for Wes	Analytical Preference of Methods for Wes
Comparative Study	M ₁	Comparative Study	M ₁	Comparative Study	M ₁
Competitive Analysis	M ₂	Competitive Analysis	M ₂	Competitive Analysis	M ₂
Prototyping	M ₃	Scenarios	M ₃	Prototyping	M ₃
Scenarios	M ₄	Parallel Design	M ₄	Scenarios	M ₄
Parallel Design	M ₅	Using Available Information	M ₅	Parallel Design	M ₅
Using Available Information	M ₆	Prototyping	M ₆	User Testing	M ₆
				Using Available Information	M ₇

III. Prerequisites for prioritization

To attain most usable software, UDCM plays vital role while development. Though there exist many UDCM, the appropriateness of their applications in varying classes of software may be investigated. Subsequent sub-sections focus on classes of software, phases of software development and usability requirements respectively as prerequisites for said prioritization.

A. Classes of Software

It has been observed that different classes of software may have distinct usability requirements [9]. There exist various types of software with different functionality and characteristics. Based on features and users of the software three different types of software classes (C_s) have been identified such as; Application Software (Aps) as C_1 , User Interface (UI) as C_2 , and Website (Wes) as C_3 .

B. Phases of Software Development

In every phase of software development, requirement of data and applicability of PDCM may vary. Mainly, four phase of software development such as Analysis (P_1); Design (P_2), Testing (P_3) and Implementation (P_4) have been considered for further study. According to preferences each phase has been assigned weights as shown in Table-2 for further computation.

Table-2: Weighing Factor Phase Wise

Phases of SDLC (P_i)	Weighing factor (W_i)	Weight Assign
P_1	W_1	4
P_2	W_2	3
P_3	W_3	2
P_4	W_4	1

C. Usability Requirements

The need of usability requirements is to specify the ease of system use, learning and efficiently the software posses for user tasks effectively [10]. Inclusion of usability concept from the early stages of software development increases developers understanding for usability requirements [3]. Each phase consists of specific usability requirements of that phase [10]. Around 32 usability requirements of P_1 have been observed such as; the end-users of the system, easy to learn, subjective satisfaction, easy to use, adaptability to the operator's skill level, understandability, operability, effective data, efficient data, flexibility, utility, environment in which people use the product, prioritize user needs, understand the need of the system etc. For P_2 , 34 Usability requirements such as; learn ability, rate of errors by users, subjective satisfaction, no mistakes, access is easy, simplicity, recoverable, stress free use, context sensitive usage, provides support, visibility, reusability, error message, reaction time, time spent on completing a particular task, time from committing an error to recovering from it, help on problems, how comfortable to use it, convenience to use it etc. have been identified. P_3 has 11 usability requirements such as; feedback, error message, primary requirements can be fulfilled or not, communicate to the users, testable, flexibility, ensuring, everything behaves as expected, tolerance, review able, self documenting, reliability etc. Total 10 usability requirements of P_4 have been observed such as simplicity, recoverable, subjective satisfaction, accepting the system, communicate to the users, user training required, tolerance, tools require carrying the task, flexibility, convenience of use etc.

IV. Prioritization of UDCM

Prerequisites for prioritization of UDCM have been described in previous section. The definitions of some useful metrics for the prioritization along with the steps of prioritization are described in this section.

1. Applicability factor (Af): Af denotes the degree to which the particular PDCM is applicable or fit to be applied during usable software development. In other words, Af is referring to the number of usability requirements satisfied by a particular PDCM out of total number of requirements defined for a phase of development and computed using expression(1).

$$Af((C_s(P_i(M_j)))) = \frac{\text{No. of usability requirements satisfied in } P_i}{\text{Total no. of usability requirements of } P_i} \quad (1)$$

Where C_s denotes the class of software, P_i represents i^{th} phase of software development and M_j is the j^{th} PDCM applicable for selected C_s . The value of s varies from 1 to 3 representing the class of software as Aps, UI and Wes respectively. Since only four phases are being considered, the value of i varies from 1 to 4. As shown in Table-1, upper bound of j is 6, 6 and 7 for class of software Aps, UI and Wes respectively.

2. Productivity factor (*Pf*): *Pf* entails about utility of a PDCM with inclusion of usability in best manner. *Pf* is the measure of efficiency and utility of the particular PDCM in a specific phase of software development. The computation of *Pf* is also performed phase wise using expression (2).

$$Pf(C_s(P_i(M_j))) = Af((C_s(P_i(M_j)))) * W_i \quad (2)$$

Where W_i is the weighing factor of the i^{th} phase of software development as denoted in Table-2.

3. Total Productivity factor (*TPf*): *TPf* is the measure of efficiency of PDCM in general i.e. in all phases of development. It is computed using expression (3) as a sum of phase wise *Pf* of each method.

Now, prioritization of UDCM can be achieved through the computation of *Af*, *Pf* and then *TPf* for a class of software. Priority to UDCM is assigned on the basis of *TPf* where higher the value of *TPf*, higher is the priority.

$$TPf(C_s(P_i(M_j))) = \sum_{i=1}^4 Pf(C_s(P_i(M_j))) \quad (3)$$

V. Case Study

Here, for each standard software class, *Af*, *Pf* and *TPf* are computed. The results of computation are shown in Table-3 and Table-4 respectively. Also, the prioritization of UDCM for standard software class based upon *TPf*, is shown in Table-5. Further, from each class two software have been studied. Usability requirements along with the general requirements of these software have been analyzed. Passing through the phases of software development, *Af*, *Pf* and *TPf* are computed for each software. The software referred for the study are mainly; Library Management System (LMS) and Hospital Management System (HMS) of class application software; Railway Ticket Generator System (RTGS) and ATM system from class of user interface and Matrimonial Website (MWes) and University Website(UWes) representing the website class of software. The results of computation of *Af* corresponding to the above mentioned software are shown in Table-6(a) and Table-6(b) whereas results representing *Pf* and *TPf* are illustrated in Table-7(a) and Table-7(b) corresponding to the same software and termed as observed values.

Further, the deviation of observed values from the standard value of *TPf* has been evaluated in terms of percentage as shown in Table-8. The deviation corresponds to the increment, decrement and no change with observed values. Considering 5% variation between standard and observed value as negligible for positive and negative differences, it has been observed that the variation has to be obvious in some cases due to domain change. There exist many factors causing this deviation and affecting usable software development as well. It includes; type of system developed, users of the system, functioning of system, domain specific system, simple requirements and usability requirements of system, and most importantly data collection methods used in different phases of software development etc.

Table-3: *Af* for standard software classes

S/W Classes Phases	Aps						UIn						Wes						
	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇
P ₁ AfP ₁ (M ₁)	0.593	0.593	0.812	0.687	0.75	0.5	0.687	0.718	0.687	0.875	0.718	0.843	0.875	0.656	0.937	0.781	0.875	0.875	0.968
P ₂ AfP ₂ (M ₁)	0.411	0.529	0.852	0.617	0.911	0.382	0.764	0.794	0.529	0.941	0.676	0.882	0.764	0.588	0.941	0.617	1	0.941	0.911
P ₃ AfP ₃ (M ₁)	0.454	0.454	1	0.727	0.636	0.363	0.636	0.636	0.545	0.636	0.363	0.727	0.636	0.818	0.545	0.727	0.909	1	0.909
P ₄ AfP ₄ (M ₁)	0.6	0.5	0.7	0.7	0	0	0.8	0.7	0.6	0	0	1	0.8	0.9	0.8	0.8	0	0.8	0

Table-4: *Pf* for standard software classes

S/W Classes Phases	Aps						UIn						Wes						
	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇
P ₁ PfP ₁ (M ₁)	2.372	2.372	3.248	2.748	3	2	2.748	2.872	2.748	3.5	2.872	3.372	3.5	2.624	3.748	3.124	3.5	3.5	3.872
P ₂ PfP ₂ (M ₁)	1.233	1.587	2.556	1.851	2.733	1.146	2.292	2.382	1.587	2.823	2.028	2.646	2.292	1.764	2.823	1.851	3	2.823	2.733
P ₃ PfP ₃ (M ₁)	0.908	0.908	2	1.454	1.272	0.726	1.272	1.272	1.09	1.272	0.726	1.454	1.272	1.636	1.09	1.454	1.818	2	1.818
P ₄ PfP ₄ (M ₁)	0.6	0.5	0.7	0.7	0	0	0.8	0.7	0.6	0	0	1	0.8	0.9	0.8	0.8	0	0.8	0
TPf(M ₁)	5.113	5.367	8.504	6.753	7.005	3.872	7.112	7.226	6.025	7.595	5.626	8.472	7.864	6.924	8.461	7.229	8.318	9.123	8.423

Table-5: Prioritization of UDCM for standard software class

Priority	Aps		UIn		Wes	
1	M ₃	Prototyping	M ₆	Prototyping	M ₆	User Testing
2	M ₅	Parallel Design	M ₄	Parallel Design	M ₃	Prototyping
3	M ₄	Scenarios	M ₂	Competitive Analysis	M ₇	Using Available Information
4	M ₂	Competitive Analysis	M ₁	Comparative Study	M ₅	Parallel Design
5	M ₁	Comparative Study	M ₃	Scenarios	M ₁	Comparative Study
6	M ₆	Using Available Information	M ₅	Using Available Information	M ₄	Scenarios
7					M ₂	Competitive Analysis

Table-6(b): Observed values of A_f

S/W Classes	Library Management System						Railway Ticket Generator System						Matrimonial Website						
	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇
P ₁ Afp ₁ (M ₁)	0.56	0.5	0.843	0.718	0.75	0.687	0.625	0.687	0.687	0.875	0.781	0.875	0.875	0.687	0.937	0.812	0.875	0.843	0.937
P ₂ Afp ₂ (M ₁)	0.558	0.588	0.852	0.588	0.941	0.470	0.705	0.735	0.529	0.941	0.647	0.882	0.735	0.529	0.941	0.647	1	0.911	0.911
P ₃ Afp ₃ (M ₁)	0.454	0.454	1	0.727	0.636	0.363	0.636	0.545	0.636	0.727	0.363	0.727	0.727	0.818	0.636	0.636	0.909	1	0.909
P ₄ Afp ₄ (M ₁)	0.5	0.5	0.9	0.7	0	0	0.8	0.7	0.6	0	0	1	0.8	0.9	0.9	0.8	0	0.8	0

Table-7(a): Observed values of P_f and TP_f

S/W	Hospital Management System						ATM System						University Website						
	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇
P ₁ Pfp ₁ (M ₁)	2.5	2.116	3.124	2.748	2.872	2.748	2.748	2.872	3	3.5	3.124	3.5	3.5	2.748	3.748	3.124	3.5	3.5	3.872
P ₂ Pfp ₂ (M ₁)	1.41	1.674	2.556	1.941	2.823	1.323	2.205	2.292	1.65	2.823	1.941	2.556	2.115	1.587	2.823	1.851	3	2.733	2.733
P ₃ Pfp ₃ (M ₁)	0.908	1.09	2	1.636	1.272	0.726	1.272	1.272	1.635	1.272	0.726	1.454	1.272	1.454	1.272	1.272	1.818	2	1.818
P ₄ Pfp ₄ (M ₁)	0.5	0.5	1	0.7	0	0	0.8	0.7	0.6	0	0	1	0.8	0.8	0.9	0.8	0	0.8	0
TPf(M ₁)	5.318	5.38	8.68	7.025	6.967	4.797	7.025	7.136	6.885	7.595	5.791	8.51	7.687	6.589	8.743	7.047	8.318	9.033	8.423

Table-7(b): Observed values of P_f and TP_f

S/W	Library Management System						Railway Ticket Generator System						Matrimonial Website						
	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇
P ₁ Pfp ₁ (M ₁)	2.24	2	3.372	2.872	3	2.748	2.5	2.748	2.748	3.5	3.124	3.5	3.5	2.748	3.748	3.248	3.5	3.372	3.748
P ₂ Pfp ₂ (M ₁)	1.674	1.764	2.556	1.764	2.823	1.41	2.115	2.205	1.587	2.823	1.941	2.646	2.205	1.587	2.823	1.941	3	2.733	2.733
P ₃ Pfp ₃ (M ₁)	0.908	0.908	2	1.454	1.272	0.726	1.272	1.09	1.272	1.454	0.726	1.454	1.454	1.636	1.272	1.272	1.818	2	1.818
P ₄ Pfp ₄ (M ₁)	0.5	0.5	0.9	0.7	0	0	0.8	0.7	0.6	0	0	1	0.8	0.9	0.9	0.8	0	0.8	0
TPf(M ₁)	5.322	5.172	8.828	6.79	7.095	4.884	6.687	6.743	6.207	7.777	5.791	8.6	7.959	6.871	8.743	7.261	8.318	8.905	8.299

Table-8: Deviation of standard and observed value

Aps						UIn						Wes						
M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇
4.09	3.63	3.81	0.55	1.28	26.14	5.98	6.68	3.02	2.40	2.93	1.51	1.21	0.77	3.33	0.44	0	2.39	1.47
4.01	0.24	2.07	4.03	0.54	23.89	1.22	1.25	14.27	0	2.93	0.45	2.25	4.84	3.33	2.52	0	0.99	0

By referring Table-8, it has been observed that in LMS incremental difference of 26.14% has been obtained for M₆. It shows that method M₆ gives the better results if the manual system exists. Similarly for HMS, incremental difference of 23.89% has been obtained. This result confirmed that method M₆ always gives the variation in results if domain of developing system is changed and it is known to developer. Rest all the methods of Aps category obtain less than 5% variation. In ATM system, incremental difference of 14.27% has been obtained for M₃. It shows that M₃ gives the better results because the working of ATM system is known to developer.

For both M₁ and M₂ more than 5% decrementing difference has been observed in case of RTGS with minor deviation of 0.98% and 1.68% respectively. Since RTGS is emerging software, we may conclude that for existing software with ample availability M₁ and M₂ showcase better results in terms of usable software development. For rest of the methods of UIn category less than 5% difference has been observed. For MWes and UWes less than 5% deviation has been observed for all methods and no deviation has been obtained in M₅ for Wes class of software i.e parallel design as obvious. After performing case study, it has been observed that priority of UDCM will be changed in case of specific software and it is shown in Table-9.

Table-9: Prioritization of UDCM

Priority	Aps			UIn			Wes		
1	M ₁	Comparative Study		M ₅	Using Available Information		M ₃	Prototyping	
2	M ₅	Parallel Design		M ₆	Prototyping		M ₅	Parallel Design	
3	M ₃	Prototyping		M ₄	Parallel Design		M ₁	Comparative Study	
4	M ₆	Using Available Information		M ₁	Comparative Study		M ₆	User Testing	
5	M ₂	Competitive Analysis		M ₂	Competitive Analysis		M ₇	Using Available Information	
6	M ₄	Scenarios		M ₃	Scenarios		M ₄	Scenarios	
7							M ₂	Competitive Analysis	

VI. Conclusion

UDCM will help to collect data in all phases of software development considering usability requirements. Prioritization of UDCM will also reduce the efforts to select appropriate data collection methods and provide ease for developer to fulfill usability requirements phase wise. As a result, developer will be able to fulfill most of the usability requirements and thus maximize user satisfaction by developing usable software. In this paper, prioritization of UDCM has been attempted for generalized software class and for specific software system as well. Further, the case study has been performed and the results verified the proposed prioritization indicating the change if domain specific software are considered for development. It has been observed that some methods result in negligible deviation whereas a particular method may deviate reasonably for different kind of software. This work recommends that generalization of UDCM may not be yet feasible completely but there remains a

scope to investigate more specific UDCM on the basis of various parameters in order to get the more usable product.

References

1. J. Nielsen, *The Usability Engineering Life Cycle*, IEEE Computer Society, Volume 25, Issue 3, pp 12-22.
2. Y.Ormeno, J.Panch, N. Fernandez and O. Pastor, *Towards a Proposal to Capture Usability Requirements Through Guidelines*, Proceedings of 7th IEEE International Conference on Research Challenges in Information Science, pp 1-12.
3. L. M. Cysneiros and A. Kushniruk, *Bringing Usability to the Early Stages of Software Development*, Proceedings of 11th IEEE International Conference on Requirements Engineering, pp 359-360.
4. Y.Tao, *Work in Progress-Introducing Usability Concepts in Early Phase of Software Development*, 35th ASEE/IEEE Frontiers in Education Conference
5. N. Bevan, N. Claridge, M. Maguire, M. Athousaki, *Specifying and Evaluating Usability Requirements Using the Common Industry Format*, Proceedings of IFIP 17th World Computer Congress, Montreal, Canada, p133-148. Kluwer Academic Publishers.
6. N. Bevan, *Usability Issues in Web Site Design*, National Physical Laboratory, Usability Services, Teddington, Middx, TW11 0LW, UK.
7. T. Geis, Dr. W. Dzida and W. Redtenbacher, *Specifying Usability Requirements and Test Criteria for Interactive Systems*.
8. R. Pressman, *Software Engineering: A Practitioner Approach*, prentice Hall, 6th edition
9. T.Tiwari, K.Paithankar, *Identifying Parameters with Data Collection Methods for Software Development in View of Usability*, 4th National Conference, INDIACom.
10. K. Paithankar, and M. Ingle, *Identification of Vital Factors by Analyzing Usability Requirements*, Second International Conference on Advances in Computer Vision and Information Technology (ACVIT 2009), Babasaheb Ambedkar Marathwada University, Aurangabad, pp 578-587.