An Intelligent Human Centred Software Development Framework

Rajeev Sharma[1], D. J. N. Singh[2]

Abstract: In the last few years, software designing and development methodologies taken attention from product delivery context to product quality and end-users' requirements satisfaction level. To, provide authentic and tangible process solution for human centric development, this research paper, provides conceptual integrated framework based on agile human centric engineering and design thinking (DT) approaches for human centred software development (HCSD) and give methods to calculate user acceptance index factors. So, the current software development come across the "human centred software development methodologies". This paper, also emphasized on the different impact factors of software process development and critically analyze user quality satisfaction ranks scale value in terms agile human centric and design thinking (DT) approaches.

1. Introduction

Software development/designing life cycle (SDLC) is a systematic approach and structured way and practiced, which always focus and emphasis to develop well define, reliable and standard software process and application according to predefined requirements and user's point of view [45i]. There are many software developments models and methodologies in software development industry. In this research paper we discussed on human centric agile engineering methodologies like Scrum, XP and design thinking (DT) approaches [1,2].Agile engineering approach is an enabler accelerate software process development and also satisfy the end-user priorities in efficient and productive manner and design thinking(DT) is an approach to understand the human expectation from the application environment and capture the mindset and attitude in respect of real operational level point of view. In software development major concern always on higher level of product usability and clarification of user requirements/demands and for this aspect/concern proposed human centred software development conceptual framework based on agile engineering and design thinking (DT) fil the gaps between human centric requirements and evaluate process designing methodologies in respect of end-users. The rest of the paper is organized as Section 2 discusses agile engineering and software development prospective approach and its impact in software development, Section 3 we discusses design thinking(DT) working environment section 4 we discusses the proposed conceptual model framework and algorithm, Section 5 we discuss the

acknowledgment, Section 6 we discuss the conclusion and future scope and section 7 we discuss the references.

2. Sustainable Software Development Through Agile Engineering:

The software designing activities becomes more complex due to the need and requirement of direct need of end-user participation in software development modeling and solutions to generate integrated software methodologies for this agile engineering play vital role in this scenario. Agile engineering work on proactive manner to adaptive manner of process designing and product development and for this agile engineering framework conceptualize divided in three working dynamics: speculation, collaboration and learning as shown in below figure 2.1 and also gives view on different phases of agile adaptive software development (AASD) life cycle and process data flow level with in development phase and figure 2.2 shows the agile engineering working environment and product life cycle and in this "A and B both are speculation sprint backlog and relationship between both level is point to point(P2P)" and upper level of this figure showing agile engineering methodologies between client and development end to within product backlog and sprint backlog phase and gives reference view on different cycles of process designing vision to end-user product delivery phase. Agile engineering provide light weight programming environment and core working principal is developers relying on technical and logical excellence of software development approaches and try to produce human centric designing and development[50-52] and make an environment for sustainable software development Agile human centric approach significantly

[1,2]*Galgotias University, Greater Noida, Uttar Pradesh, India.*
Email: hello.sharmarajeev@gmail.com, singhjn2000@gmail.com

works on develop operational level validate process, clear specification of user need, minimize ideate phase efforts, and come out testing and verification at every stages of software development.
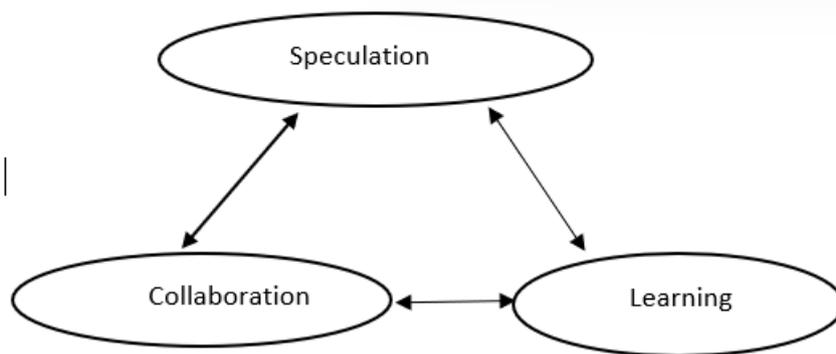

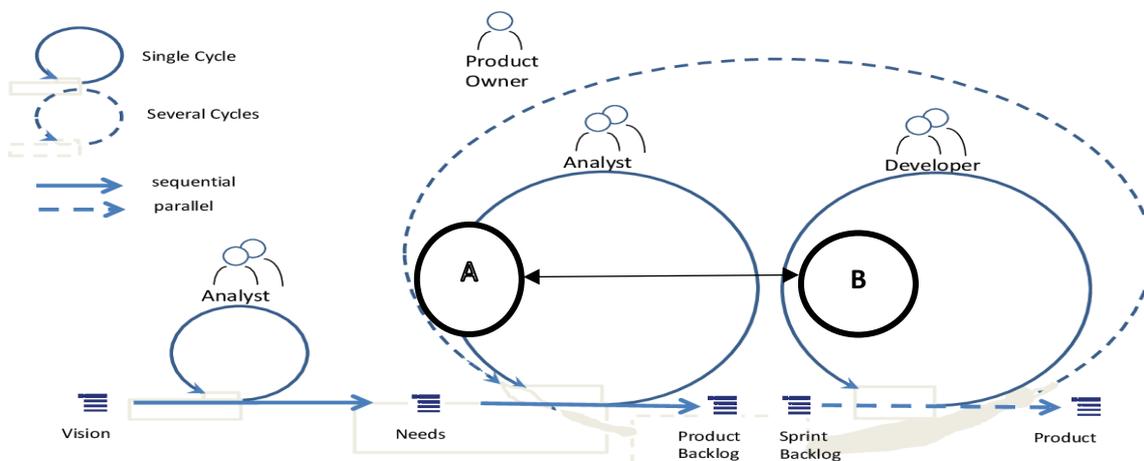**Fig 2.1**: Agile Adaptive software development (ASD) life cycle

Fig 2.2: Agile Engineering working environment and product life cycle [46-47]

2.1 Agile Engineering Human Centric Approaches:

2.2 Scrum Technology: This approach is iterative, timebox, fast feedback, rapid adaption to change, product delivery in efficient and effective manner, deliver products in short span of time, tasks are managed using backlog, regular meeting schedule to verify the scrum progress, team is self-motivated and organized[48,49] and below figure 2.3 shown the scrum approach framework and different sprint stages.
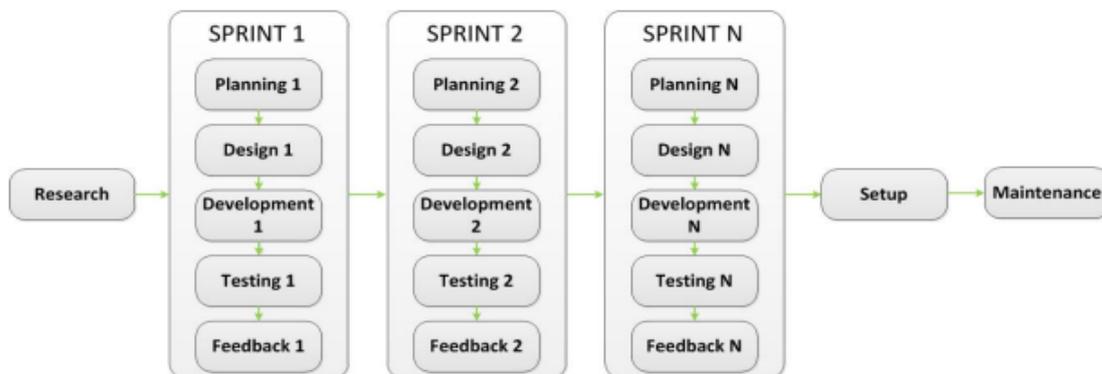

Fig2.3 Scrum approach working framework [49]

2.3 Extreme Programming:

Xtreme programming (XP) methodology suggested different thoughts and ideas of working as a package of serval aspect of innovative ideas and working principles by which software development process might be easier and convenient[53] and XP targeted generally co-located development teams in case of non-critical software development product and suggested to work for different sizes of industries world-wide[54-55] and below figure

2.4 shown customer requirements specification, iteration level and end-users feedback stages and this figure give view how user requirements come at system metaphor stage then pass through spike phase and after acceptance test and customer approval product reach to final stage "end-user feedback phase". In below table 2.1 show the different approaches of agile engineering and their impact factors in software development process management and designing and figure 2.5 shown the graphical representation of agile engineering different approaches in parallel their user acceptance level(it come down between 60-85 % of total weightage of over all software development phases) and this figure gives different stages performance level user acceptance index factors.
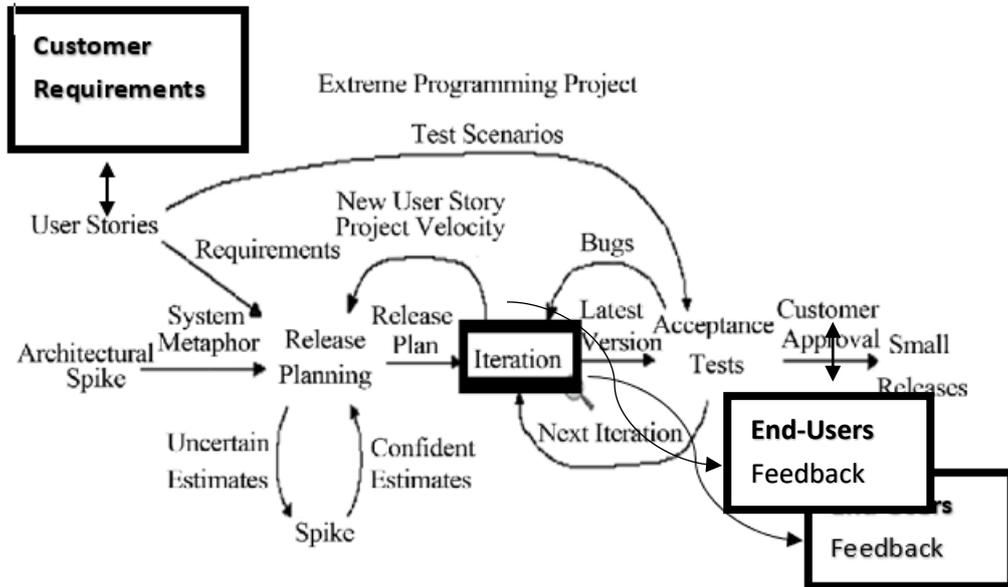

Fig 2.4 Agile Engineering Xtreme Programming End-User Environment

Table 2.1 Agile Engineering Approaches and their impact Factors [18-20]

S. No.	Agile Approaches Selection (Type)	SDLC Phases Cover in Agile Engineering	Level of SDLC Mainly Cover	Efforts (Time/Cost)	User Acceptance Level	Quality Rank (Out of 5 Scale Value)
1	Extreme Programming	Native principles and values	Phase 1-4	Middle	Given Constraints	>2<4.5
2	Agile Modeling (AM)	Address Complex Issue	Phase 1-3	Incremental Level	Focus on Group Level	>1<3.5
3	Scrum	Build a Backlog, Process development to management too	Phase 1-5	Average Level	Iteration over exhaustive	>3<4.25
4	Crystal Methodologies Family	Efficiency and habitability	Phase -3 (tools and standard declaration)	Crystal clear (Always Counted)	Project safety	>4<4.5

| 5 | Feature-Driven Development | Client value feature view | Phase 1-3 | Tangible | Timely and client view level | 34<4 |
| 6 | Adaptive Software Development | Based on rapid application development (RAD) | Phase 1-5 | Not decided | Continuous | >4<5.0 |
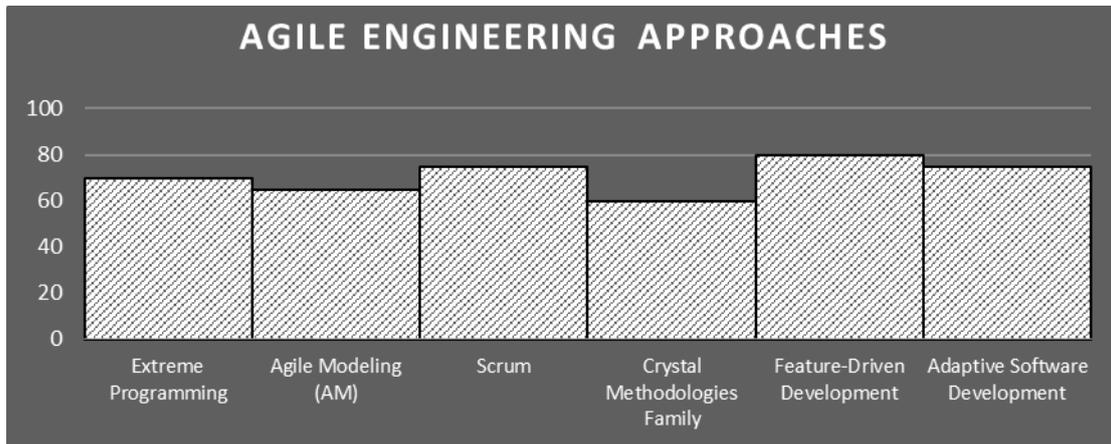
Fig. 2.5: Comparative analysis of agile engineering approaches and user acceptance level [21-22]

3. Design Thinking (Dt) Working Environment:

Design thinking (DT) provides "systematical working environmental approach for process development and management: viability, desirability/needs, and possibility/feasibility" and always concern on new understanding thoughts/ideas of innovative practices/approaches human centred process development and also acceptable within team and interest of business stakeholders and communities [3,15]. Design thinking (DT) core area of working is the innovation of new ideas/thoughts of end-user's point of view during the working environment of software process development cycle (SPDC) and generating the human-centered software process/product. The design thinking (DT) approach pulls down the application environment according to the end-user by some fundamental principles of "design thinking (DT): empathize, define, ideate, prototype, test, and end-user's operational environment. The design thinking (DT) approach gives dynamic and optimized solutions to a user's scenario problem and requirement by giving standard and quality-oriented process/product delivery and services, and also enhance the secure and significant productivity and operational betterments for business and stakeholder [3-4,17,23-24]. Some of the common activities and process stages of the design thinking (DT) approach and their relationship in all phases are shown in given below Fig.3 and this figure also gives general designing principal for human centred prototyping and iterative process solutions.
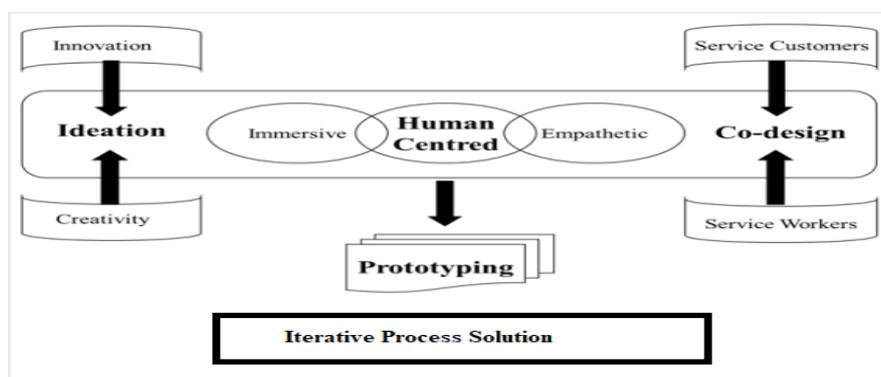
Fig 3.1 Design thinking (DT) working environment and working steps for iterative process solution for human centred software development [25-28]

3.1 Design thinking (DT) approach fundamentals core attributes

❖ Empathize: Empathize the first of design thinking (DT), where software development team real insight into users and their needs and in the focus on human-centric approach [31].

❖ Define: focus on user's point-of-view input/statement and also synthesis/analysis, problem issues and statement according the end-result and give platform for human centric development and in brief user's concern and acceptance form development end [31-33].

❖ Ideate: Point of view analysis, design problem template, brainstorming, provides solid background through quality ideas/innovation and "think outside the traditional approach of designing" and give alternative and dynamic solution of problem [20,33-34].

❖ Prototype: Design thinking (DT) this phase provides sketching, non-functional and functional prototyping solutions and development model, storyboard, and start to create different solutions and started experiment on prototyping solutions [20,34].

❖ Test: This phase provides user's feedback grid/matrix, minimum viable solution/product rollout form designing phase and match the expected and actual output to takes a mind-set of users and those requirements set in initial phase and level of previous stages of designing and development [20,32,34].

❖ so we can say "this strategy of design thinking (DT) approach give quality process designing and product development according to human centred point of view and its is open the dimension of human centred software development (HCSD)".Design thinking(DT) provide graphical prototype solution on the basis of expected outcomes and result Fig.3.2 shown the design thinking(DT) phases and user acceptance index factors(UAIE) and time index factors(TIF) that vary between 40% to 90% and time index factors count between 1 to 5.
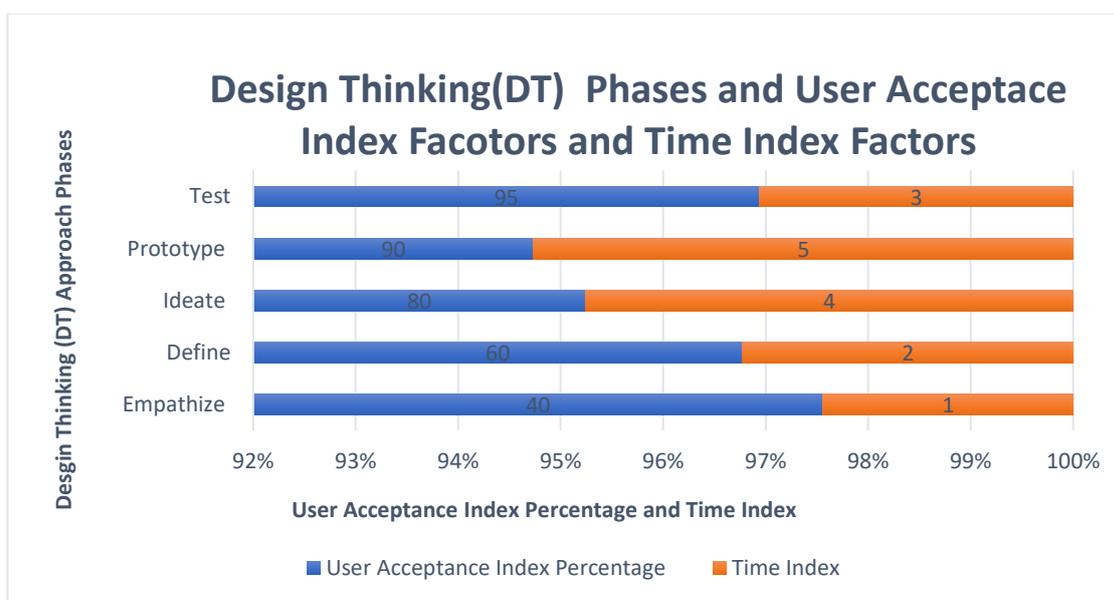

Fig. 3.2: Design thinking (DT) different approach and their user acceptance and time index

4. Human Centre Software Development Framework:

Traditional software development approaches work on client requirements and but in the current software development working environment capture the core needs and requirements specifications of users and gives quality-oriented process and product delivery and it is come true by "human centred software development"[29-31].This framework come by the "integration of agile engineering human centric approaches and design thinking (DT) techniques to develop human centred software" and core is increase end-operational behavior according to satisfaction of end-user's requirements and demands within the time limit and optimizing the solution of resources and scheduling of the project[4,32-34].This human centred software development model(HCSD) play significant role in human game theory in different social environments and dimensions[42].

4.1 Human Centred Framework Features:

The main feature is optimizing the resources and provides human oriented process development. Some major features of framework are discussing below [3-4,35-37]:

• Increase Satisfaction level of end-user.

- Feedback within software development life Cycle
- Provided conceptual integration development model concept.
- involved the real innovation in project development
- Higher level communication within team
- Product delivery and execution in validated manner
- Agile engineering approaches "Scrum and Extreme Programming" play significant role to designing this conceptual model.
- Conceptual model used "Design Thinking (DT) user's involvement technique" almost every stages of project developmentation

4.3 Framework of Human Centred Software Development Metamodel (HCSD)
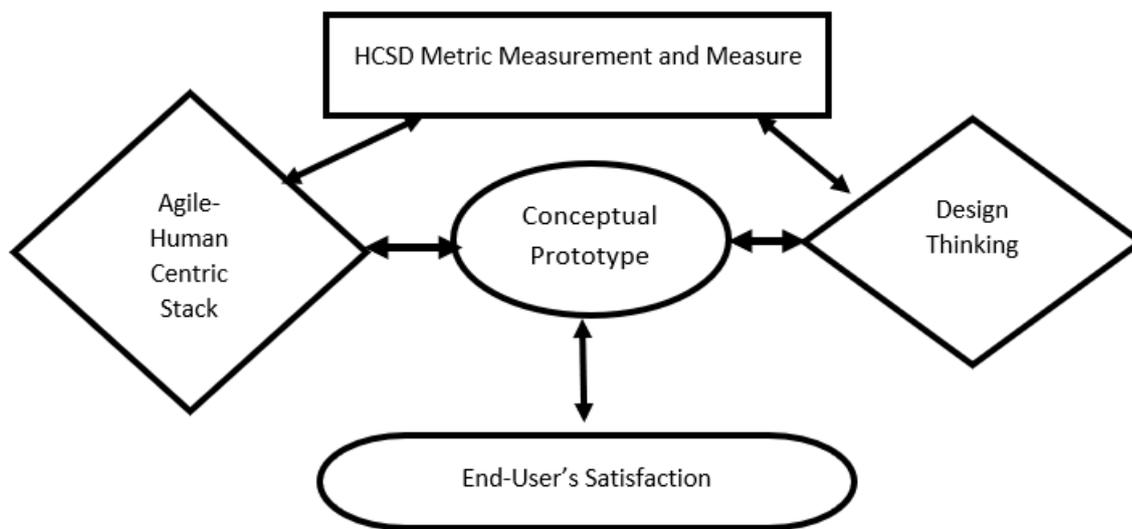
Fig. 4.1: Proposed Conceptual Meta-Model for Human Centred Software Development Framework [3-4,37-40]

In above Fig. 4.1,we discuss the framework of proposed conceptual integrated model for human centred software development (HCS) and this framework divided in five different stages/phases to developed software according to end-user's quality and requirements satisfaction point of view and before release the product we use the validation phase to enhance the quality of software/product and also increase the efficiency and usability of software designing aspects and reach the development on expected expectation level of stakeholders and users[16,24-25,39] and below Fig.4.2. given view on conceptual human centred software development user acceptance index factors in respect of traditional and human centred software development.

4.2 Algorithm of Human Centred Software Development Metamodel (HCSD):

- Input: User Requirements
- Output: Human Centred Software Development Model (HCSD)
- *Step 1:* Compute the project requirements of end-users.
- Project requirements= (S+F) * user acceptance ratio
- S=SRS requirements

- F= FRS requirements
- *Step 2:* Analyze the user feedback in middle of software development life cycle (SDLC).
- *Step 3:* Involved agile engineering XP and Scrum Sprint methods
- *Step 4:* Tested the agile engineering sprints methods according human centric requirements.
- *Step 5:* Agile engineering human centric result forward to validation phase.
- *Step 6:* Verification and validation of requirements specification done.
- *Step 7:* Ideate phase of process designing started.
- *Step 8:* Involved agile human-centric and design thinking (DT) technique.
- *Step 9:* Prototype of Human centred software development ready.
- *Step 10:* Developed system is ready for implementation & execution phase
- *Step 11:* Apply User acceptance index factors count for capture the real feedback by users.
- **Step 12:** Human-Centric environment for post feedback and responses by end-users
- End.

The user acceptance index in human centred software development model (HCSD) calculated by given formulas:

Total efforts count during traditional SDLC= (software development efforts*time)/100

Total efforts count in HCSD =∑ (efforts by agile XP and Lean Programming + Design thinking factors)/100

Total user acceptance index count percentage in HCSD model is:

∑ (Efforts/Time) *100

Total time acceptance index count is:

∑ (User Satisfactions/No of Validation failure steps) *100

Note: efforts and time count taken at the integration level of agile XP, AM and Design Thinking (DT) to reduce total efforts during process development stages and increase user acceptance index factors.
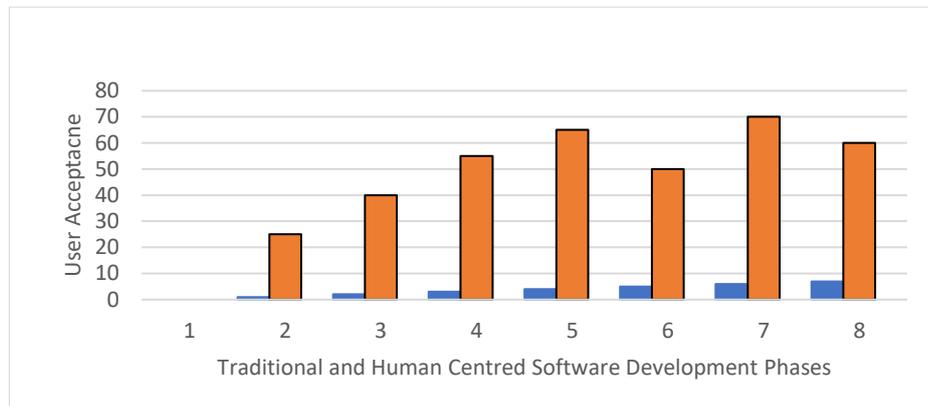
Fig. 4.2: Conceptual model user acceptance efficiency index factors in respect of traditional and human centred software development approach [39,40]

5. Acknowledgment

6. Conclusion and Future Scope

In software development practices, integrates methods that are logically inherent in the framework and process delivery techniques of software designing so in this context agile techniques gives the milestone for quality making "functional software/process" that fulfills the expected quality requirements of "end-users and satisfaction and built-up framework of human-centric software development". This conceptual framework not based on theoretical approach but involved service designing agile engineering approaches and critically examined the adoption of the design thinking (DT) approach to overcome the existing limitations in software designing and development and the area of service designing for end-users. This integrated conceptual model can find out issues/bugs and gaps in human-centric software development and co-design shell that can't be resolved by traditional software development techniques and designing methods in future we going to enhance the development phases of this proposed model and try to release more refine and sustainable human centred software development model. This purposed model gives a significant milestone in 4G human centred software development and also improve the efficiency of software development according to mindset of end-users and stakeholders too.

References

[1] Matković, P., & Tumbas, P. (2010). A Comparative Overview of the Evolution of Software Development Models. International Journal of Industrial Engineering and Management (IJIEM), 1, 163-172

[2] Documenting software systems with views VI: lessons learned from 15 Years of research & practice, Tilley Scott (2009) Proceedings of the 27th ACM International Conference on Design of Communication. SIGDOC '09, New York, pp 239–244. https://doi.org/10.1145/1621995.1622043].

[3] Human Centred Software Development Approaches, Rajeev Sharma,& J.N.Singh., 17-18 Dec. 2021,3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)

[4] Rajeev sharma, J.N. Singh, A Critical Review of Surveys Emphasizing on Agile software engineering, solid-state journal vol 64.2(2021).

[5] Cho, J. (2009). A hybrid software development method for large-scale projects: rational unified process with scrum. Issues in Information Systems, 10(2). 340-348.

[6] Empirical Research on Critical Success Factors of Agile Software Process Improvement* Jiangping Wan1,2, Ruoting Wang1, J. Software Engineering & Applications, 2010, 3, 1131-1140 doi:10.4236/jsea.2010.312132 Published Online December 2010
http://www.scirp.org/journal/jsea

[7] Working software over comprehensive documentation – Rationales of agile teams for artefacts usage Gerard Wagenaar1*, Sietse Overbeek2, Garm Lucassen2, Sjaak Brinkkemper2 and Kurt Schneide[3], Wagenaar et al. Journal of Software Engineering Research and Development (2018) 6:7 https://doi.org/10.1186/s40411-018-0051-7, Journal of Software Engineering Research and Development (2018).

[8] Dybå T, Dingsøyr T (2008) Empirical studies of agile software development: A systematic review. Information and Software Technology 50(9-10):833–859

[9] Hummel M (2014) State-of-the-Art: A Systematic Literature Review on Agile Information Systems Development. In Proceedings of the 47th Hawaii International Conference on System Sciences (HICSS), Waikoloa (HI,) USA, 6 -9 January, 2014 (pp. 4712–4721). IEEE. https://doi.org/10.1109/HICSS.2014.579

[10] https://au.targus.com/blogs/discover-targus/the-beginners-guide-to-agile-working.

[11] Matković, P., & Tumbas, P. (2010). A Comparative Overview of the Evolution of Software Development Models. International Journal of Industrial Engineering and Management (IJIEM), 1, 163-172.

[12] Wright, G. P. (2013). Success rates by software development methodology in information technology project management: A quantitative analysis, UMI Number: 3590342, UMI Dissertation Publishing, ProQuest LLC, Michigan.

[13] https://www.researchgate.net/publication/3564192 42 Designing smart dashboard system towards digital leadership in franchise organizations Article in Journal of Physics Conference Series · December 2019, Nashrullah, Muhammad, Setiawan, Ridwan, Satria, Erwinsyah, Supriatna, A, DOI: 10.1088/1742-6596/1402/2/022085

[14] Design Thinking Methods and Tools for Innovation Dimitra Chasanidou , Andrea Alessandro Gasparini2 , and Eunji Lee1 1 SINTEF ICT, Blindern, P.O. Box 124 0373 Oslo, Norway {dimitra.chasanidou,eunji.lee}@sintef.no 2 University of Oslo Library, Blindern, P.O. Box 1085 0373 Oslo, Norway a.a.gasparini@ub.uio.no.

[15] CENTERIS-International Conference on enterprise Information System/ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN/HCist 2018 Design Thinking Integrated in Agile Software Development: A Systematic Literature Review Julio Cesar Pereiraa* , Rosaria de F. S. M. Russoa a Uninove - Universidade 9 de Julho, Rua Deputado Salvador Julianelli, s/n - 1o. and, São Paulo 01156-080, Brazil

[16] Gruber M, Leon N de, George G, Thompson P. Managing by design. Acad Manag J 2015:58:1-7 Doi: http://dx.doi.org/10.5465/amj/2015.4001

[17] Review of Agile Methodologies in Software Development, Malik Hneif, Siew Hock ow, Software Engineering, University of Malaya, Kuala Lumpur, Malaysia-50603, Assoc. Prof., Department of Software Engineering, University of Malaya, Kuala Lumpur, Malaysia-50603, International Journal of Research and Reviews in Applied Sciences,ISSN:2076-734X,EISSN:2076-7366,Volume 1,Issue 1(October 2009)

[18] J. Erickson, K. Lyytinen and K. Siau, Agile Modeling, Agile Software Development, and Extreme Programming: The State of Research. In Journal of Database Management, 16(4), 2005, 88-100.

[19] Design Thinking and Agile Practices for Software Engineering: An Opportunity for Innovation, Luis Corral Monterrey Institute of Technology and Higher Education Queretaro, Mexico lrcorralv@itesm.mx, Ilenia Fronza Free University of Bozen-Bolzano Bolzano, Italy, ilenia.fronza@unibz.it Conference Paper · September 2018 DOI: 10.1145/3241815.3241864, SIGITE'18, October 3-6, 2018, Fort Lauderdale, FL, USA, https://www.researchgate.net/publication/3285466 76

[20] Hasso Plattner, Christoph Meinel, and Larry Leifer. 2010. Design thinking: understand – improve – apply. Springer.

[21] Dobrigkeit, F., de Paula, D.: Design thinking in practice: understanding manifestations of design

thinking in software engineering. In: Proceedings of the 27th ACM Joint Meeting-ESEC/FSE 2019, pp. 1059–1069 (2019).

[22] Dobrigkeit, F., de Paula, D., Carroll, N.: InnoDev workshop: a one-day introducetion to combining design thinking, lean startup and agile software development. In: 2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T), pp. 1–10. IEEE (2020)

[23] Cherry Picking - Agile Software Development Teams Applying Design Thinking Tools Franziska Dobrigkeit(B), Christoph Matthies, Philipp Pajak, and Ralf Teusner Hasso Plattner Institute, University of Potsdam, Potsdam, Germany franziska.dobrigkeit@hpi.de,oct-2021

[24] Integrating Design Thinking in Service Design Process: A Conceptual Review Tapish Panwar1*, Kalim Khan 1 Assistant Professor, Rizvi Institute of Management Studies and Research, Mumbai, India. 2 Professor, Rizvi Institute of Management Studies and Research, Mumbai, India, received: 2020/11/25, Accepted: 2021/05/27.

[25] Bazjanac, V. (1974). Architectural design theory: Models of the design process. Basic questions of design
theory, 3(1), p. 20.

[26] Rittel, H., & Webber, M. (1973). Planning problems are wicked. Polity, 4(1), p. 155-169.

[27] 24. Leonard, D., & Rayport, J. F. (1997). Spark innovation through empathic design. Harvard Business Review, 75(6), p. 102-113.

[28] Dam, R., & Siang, T. (2020). What is Design Thinking and Why Is It So Popular? Retrieved from Interaction Design - IDEO: https://designthinking.ideo.com/resources/what-is-design-thinking-and-whyis-it-so-popular.

[29] Gruber, M., De Leon, N., George, G., & Thompson, P. (2015). Managing by design. Academy of Management Journal, 58(1), P-7.

[30] The 5 Stages in the Design Thinking Process by Rikke Friis Dam, https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process.

[31] 28. Camacho, M. (2018). An Integrative Model of Design Thinking. 21st DMI: Academic Design Management Conference. London: Next Wave.

[32] https://canvas.unl.edu/courses/73802/pages/5-stages-of-design-thinking?module_item_id=1968000.

[33] Design Thinking in Practice July 2021IT Professional 23(4):95-100

[34] DIS I. 9241-210: 2010. Ergonomics of human system interaction-Part 210: Human-centred design for interactive systems. International Standardization Organization (ISO). Switzerland: 2010

[35] Wright, G. P. (2013). Success rates by software development methodology in information technology project management: A quantitative analysis, UMI Number: 3590342, UMI Dissertation Publishing, ProQuest LLC, Michigan

[36] Strode, D. E. (2006). Agile methods: a comparative analysis. In Proceedings of the 19th annual conference of the national advisory committee on computing qualifications, NACCQ (Vol. 6), 257-264].

[37] Human-Centered Software Engineering as a Chance to Ensure Software Quality Within the Digitization of Human Workflows Human-Centered and Error-Resilient Systems Development, Holger Fischer, Björn Senft, 2016, Volume 9856, ISBN: 978-3-319-44901-2

[38] Design-Thinking and User's Requirement Engineering: Human Centred Development Rajeev Sharma1, Jitendra Nath Singh2 1, 2 Galgotias University, Greater Noida, Uttar Pradesh, India, International Journal of Mechanical Engineering, ISSN: 0974-5823, Vol. 6 (Special Issue, Nov.-Dec. 2021), Copyrights @Kalahari Journals Vol. 6 (Special Issue, Nov.-Dec. 2021)

[39] How a Human-Centered Approach Impacts Software Development, July 2007DOI:10.1007/978-3-540-73105-4_8

[40] Conference: Human-Computer Interaction. Interaction Design and Usability, Xavier Ferre, Nelson Medinilla, Universidad Politécnica de Madrid, 12th International Conference, HCI ,international 2007, Beijing, China, Proceedings, Part I. July 22-27, 2007

[41] A decade of agile methodologies: Towards explaining agile software development explaining agile software development Torgeir Dingsøyr, Sridhar Nerur,Venu Gopal Balijepally Nils BredeMoea,Jouranl of systems and software, Volume 85,Issue 6,June 2012,Pages 1213-1221.

[42] MD. Ahsan Habib, K M Ariful Kabir, Jun Tanimoto "Do humans play according to the game theory when facing the social dilemma situation?" A survey study. Evergreen volume 7 issue 1(March 2020).

DOI:10.1109/MITP.2020.2993113,Sánchez-Gordón, Mary & Mendoza-Gonzalez, Ricardo & Colomo-Palacios, Ricardo. (2021).

[43] Minakshi Kar, Neha Bothra "Pandemic and Indian Education: Evolving Perspectives from Higher Education" Evergreen volume 10 issue 1(March 2023) pages_18-28.

[44] Shahroz Akhtar Khan, Ranjan Arora, Harish Kumar, Pawan Kumar Arora "A Perspective on Advances in Cloud-based Additive Manufacturing", Evergreen volume 9 issues 3 (September 2022) pages 861-869.

[45] Software Development Life Cycle Models Comparison, Consequent, Vanshika Rastogi Asst. Professor, Dept. of ISE, MVJCE Bangalore, Vanshika Rastogi / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1), 2015, 168-172, ISSN:0975-9646.

[46] Preliminary Results from a Multivocal Literature Review: 19th International Conference, PROFES 2018, Wolfsburg, Germany, November 28–30, 2018, Proceedings

[47] Agile Model Driven Development (AMDD) Scott W. Ambler Practice Leader Agile Development, IBM, XOOTIC MAGAZINE, February 2007

[48] Nigam C, Gupta S. Agile Methodology for Software Development. Inst Innov Technol Manag - J Inf Technol 2017; 3:54–64

[49] PMI. Agile Practice Guide. 2017.

[50] A Study of Agile Software Development Model (Values, Principles and Characteristics) Mrs. Gunjan Behl Assistant Professor, BVU Institute of management, Kolhapur. E-Mail – mailtogunjan@yahoo.co.in, Mr. Nripesh Kumar Nrip Assistant Professor, BVU Institute of management,Kolhapur,nripesh.nrip1985@gamil.com, International Conference on Current Trends & Challenge, in Management, Engineering, Computer Application & Technology2012

[51] Williams, L., Cockburn, A., 2003. Agile software development: it's about feedback and change. Computer 36, 39–43

[52] A decade of agile methodologies: Towards explaining agile software development explaining agile software development Torgeir Dingsøyr, Sridhar Nerur,Venu Gopal Balijepally Nils Brede Moea,Jouranl of systems and software, Volume 85,Issue 6,June 2012,Pages 1213-1221

[53] K. Beck, and C. Andres, Extreme Programming Explained: Embrace Change (2nd Edition), AddisonWesley, Boston, 2004.

[54] D. Karlström, Introducing Extreme Programming - An Experience Report. In proceedings 3rd International Conference on eXtreme Programming and Agile Processes in Software Engineering, XP 2002, Sardinia, Italy.

[55] Extreme Programming. What is Extreme Programming? [Online] Retrieved 18th March 2009. Available at: www.extremeprogramming.org ISSN: 2076-734X, EISSN: 2076-7366 Volume 1, Issue 1(October 200

[56] Dhiman, O. ., & Sharma, D. A. . (2020). Detection of Gliomas in Spinal Cord Using U-Net++ Segmentation with Xg Boost Classification. Research Journal of Computer Systems and Engineering, 1(1), 17–22. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/20

[57] Rohokale, M. S., Dhabliya, D., Sathish, T., Vijayan, V., & Senthilkumar, N. (2021). A novel two-step co-precipitation approach of CuS/NiMn2O4 heterostructured nanocatalyst for enhanced visible light driven photocatalytic activity via efficient photo-induced charge separation properties. Physica B: Condensed Matter, 610 doi:10.1016/j.physb.2021.412902

[58] Ms. Ritika Dhabalia, Ms. Kritika Dhabalia. (2012). An Intelligent Auto-Tracking Vehicle. International Journal of New Practices in Management and Engineering, 1(02), 08 - 13. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/5