Required Changes in Requirements Engineering Approaches for Socio-Cyber-Physical Systems

Ksenija Lace and Marite Kirikova

Riga Technical University, Latvia

ksenija.lace@cs.rtu.lv, marite.kirikova@cs.rtu.lv

Abstract. Requirements engineering has been mainly focused on software development, which represents relatively homogenous, stable and predictable cyber space. But even there, problems in requirements (e.g., changing, missing or irrelevant requirements) are considered as main reasons for project failure. Entering a new era of socio-cyber-physical systems, which are complex, heterogeneous systems of systems, will make requirements engineering even more challenging. Not only it is a standard practice that requirements change and evolve, and new requirements emerge frequently during the system life cycle. In socio-cyber-physical systems, requirements cannot be defined just for the cyber space, but must cover also the socio and physical spaces. There are also highly complex interrelationships, interactions and impacts between components of systems that can lead to unexpected and even unacceptable consequences in system structure and behaviour. One of the promising approaches supporting adaptability and emergency of systems is continuous requirements engineering, based on agility, flexibility and emergence. Therefore it is necessary to identify gaps in existing requirements engineering practices with respect to socio-cyber-physical systems and to propose required adjustments and enhancements in requirements engineering process.

Keywords: Requirements Engineering, Socio-Cyber-Physical System, Cyber-Physical System, Continuous Requirements Engineering.

1 Introduction

Socio-cyber-physical systems, as complex heterogeneous and adaptive emergent systems, will bring new challenges for the requirements engineering discipline [1]. These challenges will encumber requirements engineering efficiency and potentially will raise the question whether the requirements engineering is possible and useful in socio-cyber-physical systems engineering.

To answer that question, the research process was conducted in four sequential phases: (1) analysis of the "traditional" requirements engineering discipline and known issues, (2) analysis of socio-cyber-physical systems and important specifics related to requirements engineering in their context, (3) consideration of a business case of socio-cyber-physical system as an example of system characteristics influencing requirements engineering, (4) identification of required enhancements in requirements engineering approaches.

The main questions for the first phase of research were: Why the requirements engineering discipline was introduced? What is the main purpose and objectives for this discipline? What are the common practises used? Are there any issues identified for this discipline to be efficient in achieving these objectives in context of socio-cyber-physical systems?

The focus for the second phase was to describe the paradigm of socio-cyber-physical systems as well as main characteristics of such systems that can directly or indirectly impact achievement of requirements engineering objectives.

During the third phase, additional objectives for requirements engineering in socio-cyber-physical systems context were defined and for each of them the list of required actions was proposed. The proposed actions are currently just theoretical recommendations, but in further research it is intended to define more specific techniques and assess effectiveness of these techniques in real case study examples.

In the fourth phase the example of requirements engineering for one specific type of socio-cyber-physical system was investigated where defined challenges and proposed solutions were illustrated.

In conclusion, targets for further research were identified.

2 Issues of “Traditional” Requirements Engineering

Requirements engineering discipline was introduced in 1990s and refers to the process of eliciting, documenting and maintaining requirements in the systems engineering process. It is important to respect the difference between systems engineering area, where “system” means many different components of any type forming one whole, from the more narrow software engineering area, where only software components are forming a system [2], [3]. For a long time the requirements engineering discipline was mainly addressing software system development.

Fig. 1 illustrates typical relationships between software development and requirements engineering activities.

There are many software development approaches widely used, but all of them consist of the following sequential phases (Fig. 1 (1)): planning, investigation, implementation, testing, production and maintenance of the developed system [4]–[6].

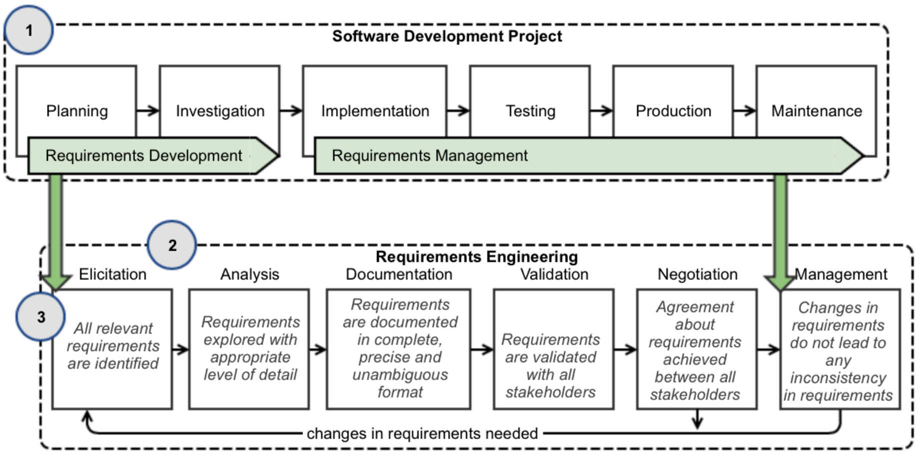

Fig. 1. Traditional Requirements Engineering Approach

Planning and investigation phases always include more or less formal and detailed requirements development activity. The main goal of this activity is to develop requirements for the system to be implemented. All further phases include requirements management activity, with the goal of ensuring requirements implementation and handling changes in requirements [4]–[6].

Requirements development and management constitute requirements engineering (Fig. 1 (2)). There are several standards for the requirements engineering process. Some of them have focus only on software development, some cover also required changes in business processes. But all of them propose that requirements engineering should include the following sequential steps and corresponding objectives (Fig. 1 (3)) [5], [7], [8]:

- Step 1 – Elicitation:
  - Activity: elicitation of requirements from different sources.
  - Objective: all relevant requirements are identified.
- Step 2 – Analysis:
  - Activity: detailed analysis of requirements and elaboration on irrelevant, incomplete, vague and inconsistent requirements.
  - Objective: requirements are explored at the appropriate level of detail.
- Step 3 – Documentation:
  - Activity: requirements documentation for communication to stakeholders.
  - Objective: requirements are documented in complete, precise and unambiguous format.
- Step 4 – Validation:
  - Activity: requirements validation from content, documentation and agreement perspectives.

  - Objective: requirements are validated with all stakeholders.
- Step 5 – Negotiation:
  - Activity: requirements communication and negotiation between involved stakeholders.
  - Objective: agreement about requirements achieved between all stakeholders.
- Step 6 – Management:
  - Activity: maintaining requirements implementation and changes.
  - Objective: changes in requirements do not lead to any inconsistency between requirements.

Effectiveness of requirements engineering is the accuracy and completeness with which defined objectives are achieved. Effectiveness of requirements engineering impacts effectiveness of software development and quality of the developed system. However, several reports [5]–[7], [9], [10] show that existing requirements engineering practices are not effective enough, as often irrelevant and poorly formulated requirements are given to the software development. It leads to requirements changes and rework.

Frequently mentioned problems in software development caused by ineffective requirements engineering are illustrated in Fig. 2 (1) [5]–[7], [9]–[12]:

- Ineffective elicitation is the reason of missing, unnecessary and irrelevant requirements.
- Ineffective analysis causes that developed system does not achieve stated goals.
- Ineffective documentation leads to incorrectly interpreted requirements.
- Ineffective validation results in the functionality of the system that cannot be effectively and efficiently used.
- Ineffective negotiation is the root cause of the significant changes in the system required after the development is done.
- Due to ineffective management changes negatively affect system's effectiveness.

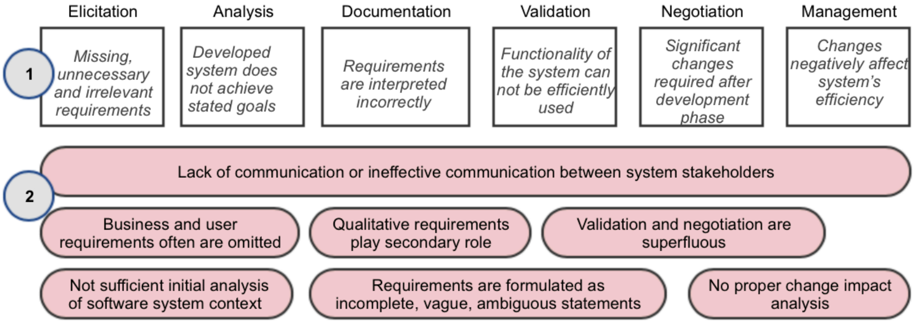

Fig. 2. Issues in Traditional Requirements Engineering

Main reasons of the ineffective requirements engineering are the following missing or low quality activities in different requirements engineering steps (Fig. 2 (2)) [5]–[7],

[9]–[12]:

- During all requirements engineering process: lack of communication or ineffective communication between system stakeholders.
- During elicitation and analysis: business and user requirements are often omitted, and not sufficient initial analysis of software system context is performed, including social and physical environments.
- During analysis and documentation: qualitative requirements (in some sources named as non-functional requirements) play secondary role, often stated in high level, vague and ambiguous format.
- During documentation and validation: requirements are formulated as incomplete, vague or ambiguous statements.
- During validation and negotiation: validation and negotiation phases often are superfluous, requirements engineering is a linear process without feedback loops.
- During negotiation and management: requirements change is a chaotic, uncontrolled process, not including proper change impact analysis.

All these issues are applicable also on requirements engineering for socio-cyber-physical systems, and should be resolved there.

3 Socio-Cyber-Physical Systems

As mentioned above, requirements engineering nowadays is used mainly in software engineering area. It means that requirements are defined mainly for the cyber space. However, the Internet of Things started a new phase in the evolution of cyber systems, where cyber space is closely integrated with physical space [1]. And recently next phase of this evolution has started, where cyber and physical spaces are intertwined with a social space. Thus we can envision that in the near future, in the majority of cases, requirements engineering will be used not just for software (cyber) systems, but rather for complex systems combining all three dimensions – socio, cyber and physical ones. [1], [13], [14].

“Socio-cyber-physical system” is a relatively new term, and its exact definition varies between different sources. This system can be seen as the intersection of the several categories of systems (see also Fig. 3) [15]–[17]:

From structure content perspective:

- Cyber-physical system, which combines mechanisms that are controlled or monitored by computer-based algorithms, tightly integrated with the Internet.
- Socio-cyber system, which combines digital structures that interact with social systems and facilitate, enhance and scale human endeavours.

From structure organisation perspective:

- Open system, which has external interactions with its environment.
- Complex system, which combines a high number of components, which form complex inter-relationships and produce nonlinear emerged behaviour.
- System of systems, which combines other systems to pool their resources and capabilities for better performance.

From behaviour perspective:

- Real time system, which is a subject to a “real-time constraint”.
- Adaptive system, which is able to respond to environmental changes or in the changes inside the system itself.
- Smart system, which uses the functions of sensing, actuation and control in order to analyse a situation, and make decisions based on the available data in a predictive or adaptive manner.

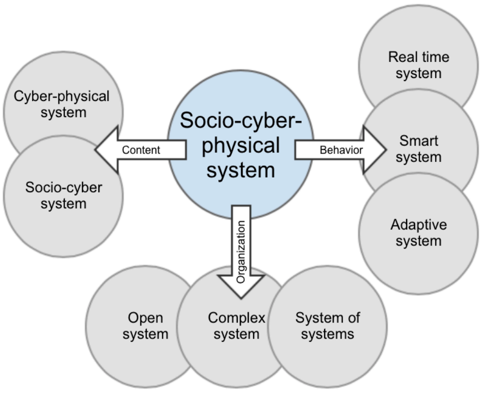

Fig. 3. Socio-Cyber-Physical System and Existing Systems Categories

From a structural perspective, a socio-cyber-physical system includes all of the following components [16], [18]–[20] (Fig. 4):

- (1) Mechanical hardware components.
- (2) Computing hardware components, that can be stand-alone components (computers) or embedded into mechanical hardware (smart devices).
- (3) Software components, that are installed on computing hardware.
- (4) Human components, that can form different social structure components.
- (5) Data and knowledge components, that are part of computing software, human or social components. Data components are used for communication and data exchange between all other components. Data can be processed, transformed into information and saved as knowledge.

The structure of socio-cyber-physical system is highly complex with many interrelationships and interdependencies between components.

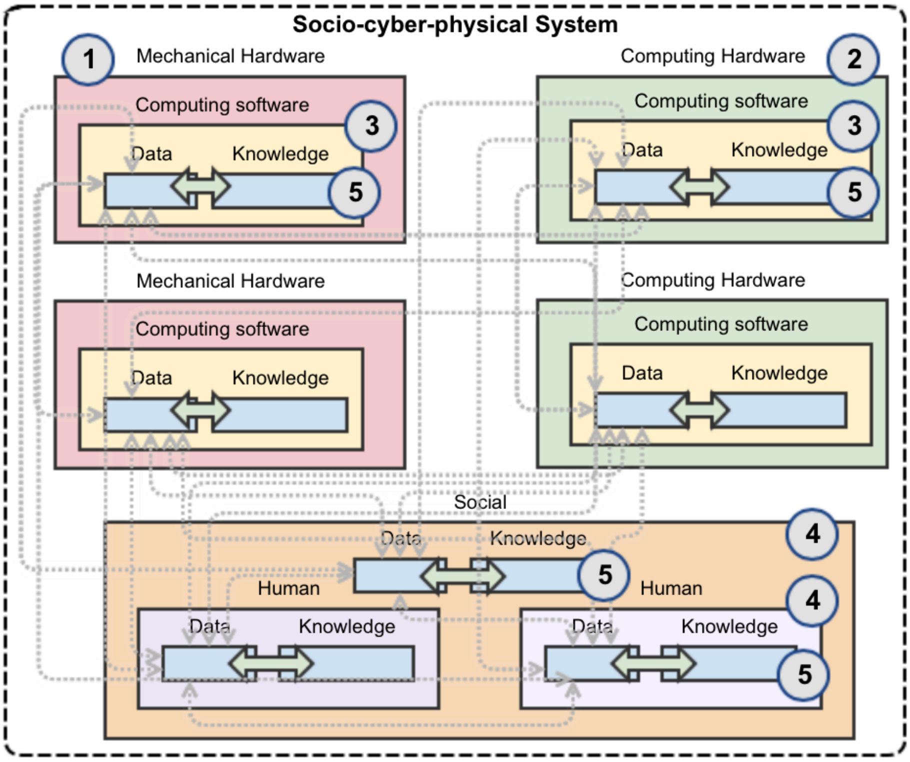

Fig. 4. Structure of Socio-Cyber-Physical Systems

Some example fields where socio-cyber-physical systems can be used are [[13], [15], [21]–[27]]:

- Operation in dangerous environments (e.g. search and rescue, firefighting).
- Exploration in inaccessible environments (e.g. deep sea, space).
- Precision operation (e.g. robotic surgery and nano-tolerance manufacturing).
- Large scale distributed coordination (e.g. traffic control, production networks).
- Situated intervention with a quick response (e.g. autonomous collision avoidance)
- Augmentation of human capabilities (e.g. assistive technologies).
- Enhancement of societal wellbeing (e.g. ubiquitous healthcare monitoring and delivery)

But these are just a few examples of numerous possible use cases.

Socio-cyber-physical system has a distinguishing set of characteristics. Some of these characteristics also apply to other types of systems, but only socio-cyber-physical system has all of them. Below are listed the characteristics that should be supported in requirements engineering grouped in two categories (systems structure and system behaviour).

Regarding system structure[14], [16], [19], [20], [15], [28]:

- Diversity (heterogeneous components from all three spaces: social, physical and cyber spaces).
- Complexity (can include other systems as components).
- Scalability (decentralized and geographically distributed components).
- Modularity (components, which can enter or leave system and emergent connections).

System behaviour [14], [16], [19], [20], [15], [28], [29]:

- Variability (system has many potential strategies for problem solving and is capable of choosing the most efficient one in specific circumstances).
- Cooperation (system involves several components in cooperative problem solving).
- Automation (system is self-sufficient and proactive in goal achievement).
- Learning (system gathers and analyses historical data, and is able to evolve).
- Prudence (system is context aware and adapts to changes in environment).
- Emergency (non-linear emergent behaviour due to complex bi-directional influence between components).
- Security (system is protected against all security risks).
- Reliability (system is highly stable and available).
- Real time (all communication and operations are subject of time constraint).

These characteristics of socio-cyber-physical systems should be supported in requirements engineering by defining requirements that require specific characteristic (e.g. system availability or response time constraints) or by defining requirements for system functionality that implements specific characteristic (e.g. learning algorithms, context analysis functions, self-monitoring processes).

4 An example of socio-cyber-physical system

One of application domains for socio-cyber-physical systems are smart cities. A smart city is a municipality that uses information and communication technologies to increase operational efficiency, share information with the public and improve both the quality of government services and citizen welfare [30]–[32]. Smart city as a complex system consists of many smart sub-systems. One of these systems is waste collection. Today, waste collection is mostly a static manual process. In smart city, waste collection should be adaptive automated process, that would analyse the context and transform itself in order to achieve objectives in the most effective and efficient way [30], [33], [32].

The main objectives of this system are providing bins free capacity according to citizens needs with as low costs as possible.

System structure is of multiple substances, complex and decentralized, for instance, we can consider the following components [31], [34], [35]:

- Mechanical hardware components: bins, trucks, dumps.

- Computing hardware components: computer hardware, mobile devices, sensors and actuators.
- Computing software: municipality services, scheduling and routing.
- Human and society components: citizens, municipality, trucks drivers, dumps workers.
- Data and knowledge components: data about bins location, capacity, contents and actual free space, data about trucks state, location, and free capacity, data about dumps location, data about available municipality services, data about configured route and schedule. Data is processed and saved as knowledge about the citizens habits, statistics about the time when bins are usually filled up quicker, history about typical time for trucks required to get from one place to another.

As can be seen, the system has not pure cyber nature, but also incorporates physical and social components. In this case requirements engineering should cover all three perspectives and requirements for physical parts (both, mechanical hardware and computing hardware) and social parts (humans, society, as well as related data and knowledge) should be defined. This requires also in-depth analysis of the interrelations and interdependencies between requirements from different perspectives.

The system should use cooperative problem solving and variable behaviour, with many potential strategies how to fulfil stated objectives, and also should be able learn from experience and find new strategies. Some strategies that can be used – finding optimal route for a truck and smart schedule of trucks during holidays. All requirements engineering activities should be focused on business goals defined for the system and clear measurable acceptance criteria. New approach required how to formulate requirements in a way that there is not one precise solution, but are several possible solutions defined by the acceptance criteria and constraints.

The system should be context aware and adapt to changes in the environment. For example, if bins get filled up quicker than usual, or trucks get broken. For this, the new types of functional requirements to be defined: context monitoring and adapting. Additionally, requirements for big data processing in real time should be defined.

The system should be scalable and modular, allowing components enter or leave the system. For example, more bins, trucks, dumps or even municipalities can be included or excluded. For supporting this, new ways to be found how to represent the model of the system structure not as a static model, but rather as a set of possible component types, components integration process and allowed interaction between components.

However, the system should operate in predefined constraints and monitor itself. For example, the system should monitor that all trucks are located in allowed areas and bins are not damaged. For this precise constraints of allowed system structure and behaviour should be defined and new types of functional requirements to be introduced: self-monitoring and correcting. Additionally, requirements validation phase should include comprehensive system behaviour simulation activity and requirements changes according to simulation results.

The system also should be protected against security risks, be stable and available. This required in-depth analysis of different kinds of quality requirements: safety, security, performance, robustness. Potentially new types of quality requirements are required.

The high number of different stakeholders, the complexity of the system, the long life cycle of the implementation and operation will lead to the changes in system requirements, potentially including changes in the business goals. This means that requirements engineering can't be performed only once prior any implementation activities, but should be integrated in all further stages. Also, requirements engineering should become iterative process, with several sequential phases, each of them validating results of the previous phases, implementing changes in existing requirements, removing unneeded requirements and adding new requirements. For this new approach for managing requirements is required, perceiving changes in requirements not as a which would allow change requirements in more efficient and qualitative way.

5 Required Changes in Requirements Engineering for Socio-Cyber-Physical Systems

For supporting engineering of socio-cyber-physical systems, requirements engineering activities should (1) improve requirements engineering effectiveness and (2) ensure previously defined characteristics of socio-cyber-physical systems.

For requirements engineering effectiveness, additional activities or improved quality of existing activities is required (Fig. 5).

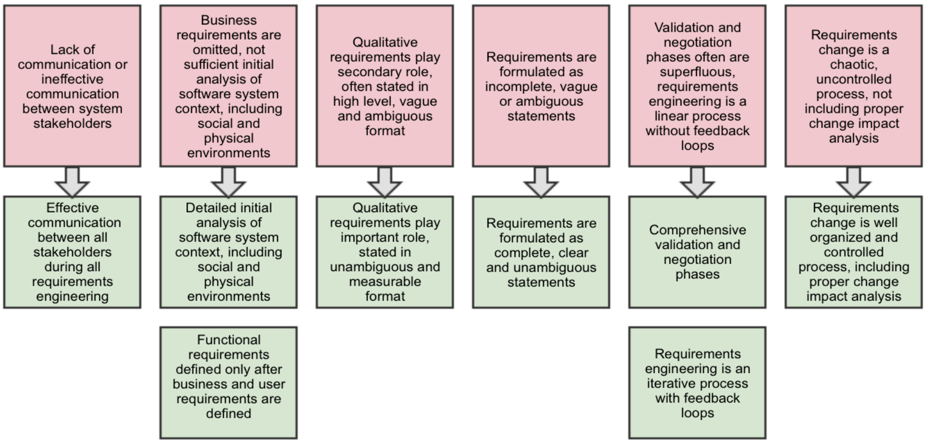

Fig. 5. Required Enhancements for Requirements Engineering Effectiveness

In particular, the following enhancements in requirements engineering are required [9]–[12]:

- Effective and sufficient communication between all system stakeholders during all requirements engineering process.
- Detailed initial analysis of software system context, including social and physical environments.
- Functional requirements defined only after business and user requirements are defined.
- Qualitative requirements play an important role, stated in unambiguous and measurable format.
- Requirements are formulated as complete, clear and unambiguous statements.
- Comprehensive validation and negotiation phases.
- Requirements engineering is an iterative process with feedback loops.
- Requirements change is well organized and controlled process, including proper change impact analysis.

For supporting characteristics of socio-cyber-physical systems, additional activities in the scope of requirements engineering or more effort in the existing activities are required (Fig. 6, (1) – system structure, (2) – system behaviour).

Identified requirements engineering activities for the waste management system example grouped by different characteristics of socio-cyber-physical systems, which are supported by these activities. Following activities in requirements engineering were identified (see Fig. 7) [12], [17], [36]–[46]:

- Due to diversity, extend the requirements scope to cover all three dimensions: cyber, physical and social dimensions. Existing requirements engineering practices to be tested for physical and social dimensions, and new practices / adjustments in existing practices to be proposed.
- For scalability and modularity, build a model for system structure, consisting of interacting components. And due to an emergency, allow these components to perform and interact by defined rules for stated business goals and observe emerging behaviours.
- Due to the modularity and emergency, define how the system can monitor itself and how it should perform in case of any failures.
- Due to the complexity and for learning, be prepared that we will not be able to define initially all requirements in a precise and detailed format therefore iterate requirements engineering activities.
- Due to the complexity and emergency define how different requirements can impact each other and to what consequences it will lead.
- For variability and cooperation, focus on business requirements and goals for the system, rather than on functional system requirements.
- For variability and automation, define requirements in such manner, that there are several possible correct solutions for system structure and behaviour, depending on the context. Additionally, requirements how system will detect and handle changes in the environment should be defined.

- For learning and prudence, prepare the system for analysing huge volume of data and learn through this. Additionally, teach a system how to adapt and allow the system to achieve business goals in the most efficient way, so that system is able to select the most efficient strategy and even is able to introduce new strategies.
- For learning, variability, automation and emergency, focus on defining constraints for allowed system behaviour rather than defining behaviour itself.
- Due to an emergency, simulate the system behaviour, detect possible negative scenarios and adjust requirements before implementation. In each socio-cyber-physical system each moment numerous events, decisions and actions may happen. There are many possible futures, only one of them will happen, but we need to be prepared for as many as possible of them.
- For automation, security, reliability and real time, define comprehensive requirements in a measurable and precise way, especially: real-time responsiveness, robustness, reliability, security. So that system performs according to stated quality standards, regarding of external impacts.

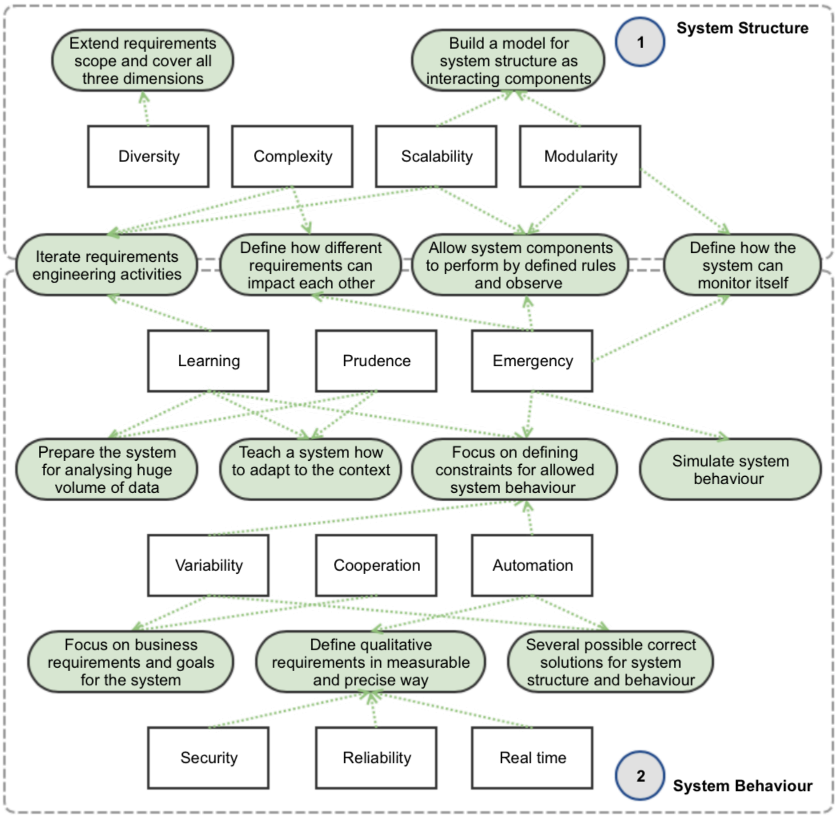

Fig. 6. Required Activities in Requirements Engineering for Socio-Cyber-Physical Systems

All identified characteristics require continuous requirements engineering, based on agility, flexibility and emergence.

The unique combination of all activities for requirements engineering effectiveness and activities required for supporting characteristics of socio-cyber-physical systems (Fig. 7 (2)) mapped to the appropriate requirements engineering steps (Fig. 7 (1)) defines main focus areas during the requirements engineering process for socio-cyber-physical systems. All these activities should be planned as mandatory and cannot be omitted for effective requirements engineering in socio-cyber-physical systems development.

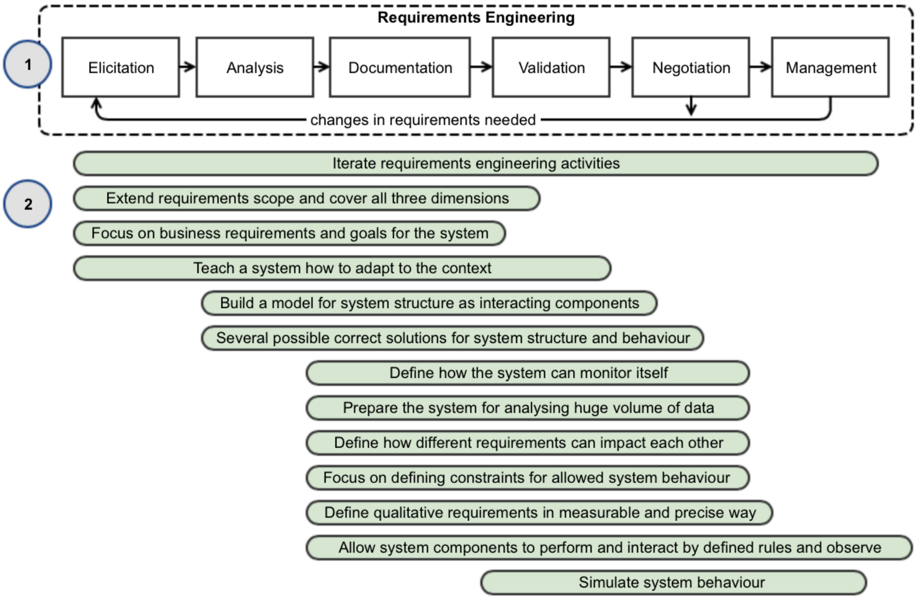

Fig. 7. Additional Activities in Requirements Engineering Process

6 Conclusions

Cyber systems evolution is entering new era of socio-cyber-physical systems. These systems have highly complex heterogeneous structure with many interrelationships between components, and context aware adaptive and learning behaviour.

In order to be able to build such systems, comprehensive requirements should be developed and well managed. Requirements development and management are executed during requirements engineering activity in the system development process.

But today the requirements engineering discipline still mainly focuses on cyber part of the systems and the scope of software system development projects. Also, there are still open discussions about the effectiveness of existing requirements engineering approaches.

In this paper main existing issues in requirements engineering, that already cause problems in software development projects, were reviewed and required enhancements in requirements engineering were identified.

From another side, socio-cyber-physical systems were introduced, main characteristics of these systems were stated and additional activities in requirements engineering for supporting these characteristics were listed.

Activities for improving requirements engineering activities and for supporting characteristics of socio-cyber-physical systems were mapped to requirements engineering steps, illustrating main focus areas in the requirements engineering process.

In this work, required enhancements are still formulated as high-level activities, and future work will revise proposed activities in more detail, elaborate specific techniques, models and artefacts that can be used. Future direction of research is to investigate existing continuous requirements engineering approaches, methods, models and tools, and assess their impact on requirements engineering for socio-cyber-physical systems through applying for case study projects.

References

[1] H. Zhuge, “Cyber-physical society-the science and engineering for future society,” *Futur. Gener. Comput. Syst.*, vol. 32, no. C, pp. 180–186, Mar. 2014.

[2] S. A. Sheard, “Systems engineering for software and hardware systems: point-counterpoint,” *INCOSE Int. Symp.*, vol. 8, no. 1, pp. 928–936, 1998.

[3] S. Koolmanojwong and B. Boehm, “Educating software engineers to become systems engineers,” in *2011 24th IEEE-CS Conference on Software Engineering Education and Training (CSEE&T)*, 2011, pp. 209–218.

[4] P. M. Huang, A. G. Darrin, and A. A. Knuth, “Agile hardware and software system engineering for innovation,” in *2012 IEEE Aerospace Conference*, 2012, pp. 1–10.

[5] P. F. Katina, C. B. Keating, and R. M. Jaradat, “System requirements engineering in complex situations,” *Requir. Eng.*, vol. 19, no. 1, pp. 45–62, Mar. 2014.

[6] F.-L. Li, J. Horkoff, L. Liu, A. Borgida, G. Guizzardi, and J. Mylopoulos, “Engineering requirements with Desiree: an empirical evaluation,” *Adv. Inf. Syst. Eng.*, vol. 9694, pp. 221–238, May 2016.

[7] K. Schmid, “Challenges and solutions in global requirements engineering - a literature survey,” in *Software Quality. Model-Based Approaches for Advanced Software and Systems Engineering*, 2014, pp. 85–99.

[8] J. M. Fernandes and R. J. Machado, “Requirements engineering,” in *Requirements in Engineering Projects*, Springer International Publishing, 2016.

[9] R. N. Memon, S. S. Salim, and R. Ahmad, “Analysis and classification of problems associated with requirements engineering education: towards an integrated view,” *Arab. J. Sci. Eng.*, vol. 39, no. 3, pp. 1923–1935, Mar. 2014.

[10] R. E. M. Champion, “Assessing the quality of requirements engineering products,” in *Software Quality and Productivity: Theory, practice, education and training*, M. Lee, B.-Z. Barta, and P. Juliff, Eds. Boston, MA: Springer US, 1995, pp. 215–218.

[11] K. M. Adams, “Introduction to non-functional requirements,” in *Nonfunctional Requirements in Systems Analysis and Design*, Cham: Springer International Publishing, 2015, pp. 45–72.

[12] A. Ebert, S. R. Humayoun, N. Seyff, A. Perini, and S. D. J. Barbosa, “Bridging the gap between requirements engineering and human-computer interaction,” in *Usability- and Accessibility-Focused Requirements Engineering*, 2016, vol. 9312, pp. 3–7.

[13] B. Sokolov, R. Yusupov, D. Verzilin, I. Sokolova, and M. Ignatjev, “Methodological basis of socio-cyber-physical systems structure-dynamics control and management,” in *Digital Transformation and Global Society*, 2016, pp. 610–617.

[14] S. Balandin, S. Andreev, and Y. Koucheryavy, “OLD Internet of things, smart spaces, and next generation networks and systems: 15th international conference, NEW2AN 2015 and 8th conference, ruSMART 2015 St. Petersburg, Russia, august 26-28, 2015 proceedings,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9247, pp. 80–94, 2015.

[15] D. Moldovan, G. Copil, and S. Dustdar, “Elastic systems: towards cyber-physical ecosystems of people, processes, and things,” *Comput. Stand. Interfaces*, vol. 57, pp. 76–82, 2018.

[16] I. Horvath, “Beyond advanced mechatronics: new design challenges of social-cyber-physical systems,” in *ACCM-Workshop on „Mechatronic Design“*, 2012, pp. 1–20.

[17] S. Rho, A. V Vasilakos, and W. Chen, “Cyber physical systems technologies and applications,” *Futur. Gener. Comput. Syst.*, vol. 56, pp. 436–437, 2016.

[18] E. Mueller, X.-L. Chen, and R. Riedel, “Challenges and requirements for the application of industry 4.0: a special insight with the usage of cyber-physical system,” *Chinese J. Mech. Eng.*, vol. 30, no. 5, pp. 1050–1057, Sep. 2017.

[19] H. Fleischmann, J. Kohl, and J. Franke, “A reference architecture for the development of socio-cyber-physical condition monitoring systems,” in *2016 11th System of Systems Engineering Conference (SoSE)*, 2016, pp. 1–6.

[20] J. Zeng, L. T. Yang, M. Lin, H. Ning, and J. Ma, “A survey: cyber-physical-social systems and their system-level design methodology,” *Futur. Gener. Comput. Syst.*, 2016.

[21] R. Vroom and I. Horvath, “Cyber-physical augmentation – an exploration,” in *Tools and Methods of Competitive Engineering (TMCE)*, 2014.

[22] M. R. Lowry, “Software construction and analysis tools for future space missions,” in *Tools and Algorithms for the Construction and Analysis of*

Systems: 8th International Conference, TACAS 2002 Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2002 Grenoble, France, April 8--12, 2002 Proceedings, J.-P. Katoen and P. Stevens, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 1–19.

[23] H. Liu *et al.*, "A review of the smart world," *Futur. Gener. Comput. Syst.*, 2017.

[24] B. Ioan, C. V. Tecaru Berekmeri, and I. Blebea, "Considerations regarding the integration-intrication processing the nature and technology," *ACTA Univ. Cibiniensis*, vol. 64, no. 1, p. 82, 2014.

[25] A. Sheth, P. Anantharam, and C. Henson, "Physical-cyber-social computing: an early 21st century approach," *IEEE Intell. Syst.*, vol. 28, no. 1, pp. 78–82, 2013.

[26] J. Ma *et al.*, "Cybermatics: a holistic field for systematic study of cyber-enabled new worlds," *IEEE Access*, vol. 3, pp. 2270–2280, 2015.

[27] H. Ning, H. Liu, J. Ma, L. T. Yang, and R. Huang, "Cybermatics: cyber–physical–social–thinking hyperspace based science and technology," *Futur. Gener. Comput. Syst.*, vol. 56, pp. 504–522, 2016.

[28] H. Zhuge, "Semantic linking through spaces for cyber-physical-socio intelligence: A methodology," *Artif. Intell.*, vol. 175, no. 5, pp. 988–1019, 2011.

[29] H. Muccini, M. Sharaf, and D. Weyns, "Self-adaptation for cyber-physical systems: a systematic literature review," in *2016 IEEE/ACM 11th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, 2016, pp. 75–81.

[30] H. Kopackova and P. Libalova, "Smart city concept as socio-technical system," in *2017 International Conference on Information and Digital Technologies (IDT)*, 2017, pp. 198–205.

[31] C. Kyriazopoulou, "Smart city technologies and architectures: a literature review," in *2015 International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, 2015, pp. 1–12.

[32] A. Król, P. Nowakowski, and B. Mrówczyńska, "How to improve WEEE management? Novel approach in mobile collection with application of artificial intelligence," *Waste Manag.*, vol. 50, pp. 222–233, 2016.

[33] C. G. Cassandras, "Smart cities as cyber-physical social systems," *Engineering*, vol. 2, no. 2, pp. 156–158, 2016.

[34] A. Arroub, B. Zahi, E. Sabir, and M. Sadik, "A literature review on smart cities: paradigms, opportunities and open problems," in *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, 2016, pp. 180–186.

[35] A. Smirnov, A. Kashevnik, and A. Ponomarev, "Multi-level self-organization in cyber-physical-social Systems: smart home cleaning scenario," *Procedia CIRP*, vol. 30, pp. 329–334, 2015.

[36] H. Fleischmann, J. Kohl, and J. Franke, "A modular web framework for socio-CPS-based condition monitoring," in *2016 IEEE World Conference on Factory Communication Systems (WFCS)*, 2016, pp. 1–8.

[37] T. Nguyen, "A modelling & simulation based engineering approach for socio-

cyber-physical systems," in *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, 2017, pp. 702–707.

[38] G. Lenzini, S. Mauw, and S. Ouchani, "Security analysis of socio-technical physical systems," *Comput. Electr. Eng.*, vol. 47, pp. 258–274, 2015.

[39] D. Amyot, A. A. Anda, M. Baslyman, L. Lessard, and J. M. Bruel, "Towards improved requirements engineering with SysML and the user requirements notation," in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, 2016, pp. 329–334.

[40] E. C. Groen, J. Doerr, and S. Adam, "Towards crowd-based requirements engineering a research preview," in *Requirements Engineering: Foundation for Software Quality: 21st International Working Conference, REFSQ 2015, Essen, Germany, March 23-26, 2015. Proceedings*, S. A. Fricker and K. Schneider, Eds. Cham: Springer International Publishing, 2015, pp. 247–253.

[41] A. L. Szejka, A. Aubry, H. Panetto, O. C. Júnior, and E. R. Loures, "Towards a conceptual framework for requirements interoperability in complex systems engineering," in *On the Move to Meaningful Internet Systems: OTM 2014 Workshops*, 2014, pp. 229–240.

[42] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security - a survey," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1802–1831, 2017.

[43] G. Biamino, "So smart - modeling social contexts to improve smart objects awareness in pervasive computing environments," in *2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011, pp. 393–394.

[44] G. Biamino, "Modeling social contexts for pervasive computing environments," in *2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011, pp. 415–420.

[45] G. Biamino and F. Cena, "Social awareness and user modeling to improve objects intelligence," in *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, 2011, vol. 3, pp. 118–121.

[46] S. Rho, A. V Vasilakos, and W. Chen, "Cyber–physical systems technologies and application — part II," *Futur. Gener. Comput. Syst.*, vol. 61, pp. 83–84, 2016.