
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Pikas, Ergo; Koskela, Lauri; Oehmen, Josef; Dave, Bhargav
From Checklists to Design Process Support Systems

Published in:
Proceedings of the 27th Annual Conference of the International Group for Lean Construction

DOI:
[10.24928/2019/0173](https://doi.org/10.24928/2019/0173)

Published: 01/01/2019

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Pikas, E., Koskela, L., Oehmen, J., & Dave, B. (2019). From Checklists to Design Process Support Systems: Initial Framing. In *Proceedings of the 27th Annual Conference of the International Group for Lean Construction* (pp. 83-96). (Proceedings of the Annual Conference of the International Group for Lean Construction). International Group for Lean Construction (IGLC). <https://doi.org/10.24928/2019/0173>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

FROM CHECKLISTS TO DESIGN PROCESS SUPPORT SYSTEMS: INITIAL FRAMING

Ergo Pikas¹, Lauri Koskela², Josef Oehmen³, and Bhargav Dave⁴

ABSTRACT

Building project delivery is beset with many long-standing problems. Often, these problems, resulting in failures of facilities and cost-time overruns, are directly related to poor design and design management practices. This motivated the definition of the main aim to develop an initial framing for the design process support systems, incorporating ideas from the human error and performance management domains, and on checklists. In this conceptual paper, a literature review method is used. It is suggested that cognitive systems engineering could be used to conceptualize the designers work and to incorporate checklists into the design process. Then, key aspects and elements for the development of design process support systems are addressed.

KEYWORDS

Error management, checklist, design process, design support systems.

INTRODUCTION

Problems and accidents plague the delivery of construction projects. According to (Eurostat 2018), out of 3,876 fatal accidents at work in the 28 EU countries during 2015, 21% took place in the construction sector. Many accidents have been directly or indirectly related to the design factors. (Behm 2005) analyzed 224 fatal construction accidents from the National Institute for Occupational Safety and Health (NIOSH) Fatality Assessment Control and Evaluation (FACE) database and found that 42% of the accidents were associated with design factors.

Design errors have been considered the primary “contributor to building and infrastructure failures as well as project time and cost overruns” (Lopez et al. 2010). In the construction industry, checklists in the form of, for example, schedules (although not usually viewed as a checklist), templates, and review guidelines for organizing and managing the design activity have been used. These instruments have been theorized

¹ Researcher, Department of Technology, Management and Economics, Technical University of Denmark, Denmark, ekas@dtu.dk

² Professor, University of Huddersfield, Department of Architecture and 3D Design, UK, L.Koskela@hud.ac.uk

³ Associate Professor, Department of Technology, Management and Economics, Technical University of Denmark, Denmark, jooehm@dtu.dk

⁴ Senior Researcher, Department of Computer Science, Aalto University, Finland, bhargav.dave@aalto.fi

mainly from the managerial or organization perspectives but not necessarily from the human error and performance perspective.

In this paper, the aim is to develop an initial framing for the design process support systems based on the theoretical and the practical ideas and concepts related to human error and performance management and the use of checklists in aviation and design domains. Literature review method is used to clarify the underpinning ideas. Overall, this paper is divided into two major parts. In the second part, key aspects of the design support systems are discussed: product versus process-based support systems, central objects, and elements of support systems, process monitoring and the role of checklists in support systems.

HUMAN ERROR AND CHECKLISTS

In this section, the philosophy, approaches, and types of human error management are addressed first, and checklists devised in different industries to address human error and performance are addressed second. Finally, the main points are discussed.

HUMAN ERROR AND HUMAN ERROR MANAGEMENT

In 1935, at Wright Air Field in Dayton, Ohio, the US Army Corps held a competition that was supposed to be a formality to select the military's next-generation bomber. However, Boeing's Model 299, which was seen to be superior, crashed in a blazing explosion, killing two of the five crew members. The investigation revealed that the crash was caused by a 'pilot error' (Gawande 2010). The solution to the problem was not the redesign of the airplane or more training, but a simple pilot's checklist.

Furthermore, since the chemical plant explosion in Flixborough (1974), the failure of the second nuclear reactor on Three Mile Island (1979), and the explosion of the reactor's core in Chernobyl (1986), human errors and organizational failures have become to be considered as the primary contributors to accidents (Reason 1990). Nowadays, research on human error can be found in several disciplines, including construction (Behm 2005; Saurin et al. 2008).

Philosophy of Human Error

The philosophers (Gorovitz and MacIntyre 1975) addressed the nature of human fallibility and described two sources for "why humans fail in what they set out to do in the real world". The first source is the necessary fallibility: humans are not omniscient. In productive goal-directed activities, there are always particulars of a situation that cannot be reduced to the law-like generalizations and initial conditions. Even the best possible judgment can turn out to be erroneous (Gorovitz and MacIntyre 1975). The second source is the ineptitude, or human error, with the various degrees of seriousness, caused by the failure to apply existing knowledge (Gorovitz and MacIntyre 1975).

(Senders and Moray 1991) defined the human error as a deviation from intention, desire or expectations, from something that was "not intended by the actor; not desired by a set of rules or an external observer; or that led the task or system outside its acceptable limits". However, this definition has limited value in the context of design, where ends and means, as well as the design process, are mutually dependent and in constant flux. Furthermore, creativity is at the core of designing, involving routine and non-routine, subject- and object-

oriented activities (Love 2002). This implies that in design, it is rather difficult to distinguish between the erroneous and successful activity. (Woods et al. 1994) argued that ascribing “error to the actions of some person, team, or organization is fundamentally a social and psychological process”.

Similarly, (Hollnagel and Amalberti 2001) argued that the dichotomization of human activity either ‘correct’ or ‘error’ is an oversimplification of a complex phenomenon. Instead of looking for “human errors” as either causes or events, human errors need to be considered as part of the normal range of human performance with various degrees of variability. Particularly, (Hollnagel and Amalberti 2001) proposed that management should “[...] find where performance may vary, how it may vary, and how the variations may be detected and – eventually – controlled”. Also, to find out why and how things go right and amplify it.

Due to the complex nature of design (Lindemann et al. 2009), design activity being subject to uncertainties, tradeoffs, and emergencies, amplified by time pressure, irregular demand, and overcrowding, there will be a gap between ‘work-as-imagined’ (e.g., by managers who dictate procedures) and ‘work-as-done’ (Wachs and Saurin 2018). That is, actual work situations do not comply with pre-defined plans and procedures. According to (Suchman 1987), plans and procedures, as well as social and material environments, are better framed as “resources for action”. A design agent and designing cannot be isolated from their context (Ullman 2002).

Approaches to Error Management

(Reason 1990) argued that there are two approaches for human error management: (1) the person approach and (2) the system approach. The person and system approaches subscribe to different views of error causation and philosophies of error management, and thus, have different practical implications.

In the person approach, the focus is on individual errors. However, according to (Reason 2000), the person approach has a significant limitation: focusing on individuals isolates errors from their context, which leads to the failure to identify the common patterns. In the system approach, in addition to human errors, the focus is on the environmental conditions of human work and the development of means to avert errors or mitigate their effects. It is assumed that humans are fallible and errors will occur. However, these errors are seen as consequences rather than causes, with their origin in systemic factors, “[including] recurrent error traps and organizational processes that give rise to them” (Reason 2000).

A more recent approach, named cognitive systems engineering, has been adopted to describe and analyze complex man-machine ensembles (Hollnagel and Woods 2005). A cognitive system is defined as “an adaptive system which functions using knowledge about itself and the environment in the planning and modification of actions” (Hollnagel and Woods 1999). In cognitive systems engineering, instead of operating on the level of physical or physiological, the focus is on the level of cognitive functions (Woods and Hollnagel 2006). That is, the emphasis is on the “joint cognitive systems, where human-machine are treated as interacting cognitive systems” (Hollnagel and Woods 2005). These interactions between the human (cognition) and its environment, including social, material

and cognitive structures (such as conceptual symbolic artefacts; e.g., building information modelling), and how agents regulate and control their behavior and performance are the object of study and unit of analysis, respectively.

Artefacts, either tools or prosthesis, form a central constituent in the joint cognitive systems (Hollnagel 2002). According to (Hollnagel and Woods 2005), the artefact is “something made for a specific purpose”. The distinction, whether the artefact is a tool or prosthesis, is dependent on the way it is used. That is, whether it is developed to enhance the user’s abilities to perform tasks and solve problems (e.g., design decision support systems) or replace certain functions (e.g., wheelchair).

Levels of Cognitive Involvement and Types of Errors

It is well established that human performance involves different levels of control, a central premise in activity theory (Bedny and Meister 2014). Several concepts and models for theorizing on human performance and control of behavior exist, such as the distinction between the micro-, macro-, and metacognition (Klein et al. 2003; Woods 2009). However, probably one the most well-known, due to its longevity, breadth of use, and success, is the Rasmussen’s Skill-Rules-Knowledge (S-R-K) model. It was initially proposed at the end of the 1970s and has served the human factors research community since (Woods 2009).

However, the S-R-K model developed based on the information processing view of cognition, pioneered by Newell and Simon, has been criticized in the cognitive systems engineering domain (Hollnagel 2002; Hollnagel and Woods 2005). The main criticism is that it isolates the human cognition from its environment (social, material and cognitive structures). Consequently, the overall system perspective was somewhat lost. In cognitive systems engineering, to overcome this limitation, the focus moved from internal functions and structures of human or machine to the external joint cognitive systems (Hollnagel 2002).

The different human performance levels have been associated with different types of errors. The development of an error taxonomy is the most common approach to transforming the theories of human error into a usable form (Senders and Moray 1991). (Reason 1990) proposed that the types of errors include slips, lapses, and mistakes. Slips refer to the attention failure of carrying out unintended or unplanned action(s) and lapses to the memory failure of omitting intended or planned actions. Mistakes are the result of using wrong rules, incorrect application of rules or failure to apply the correct rule; or the insufficient or incorrect knowledge or misapplication of existing knowledge to new situations. Finally, violations are deliberate deviations from standards, safe operating practices, and procedures (Reason 1990).

CHECKLISTS IN HUMAN ERROR/PERFORMANCE MANAGEMENT

Various artefacts to avoid human error and assure the performance of designers have been developed, including checklists to increase the quality of outcomes and to reduce the risk of costly mistakes. For example, human error management and pilot’s checklists are arguably the cornerstones of operational aviation safety. Although checklists might look simple, they are complex socio-technical interventions and need to be designed, developed

and implemented consciously (Gawande 2010). In the following, the theoretical and the practical ideas and concepts related to the use of checklists are addressed.

General Concepts on Checklists

(Scriven 2000) defined the checklist as “a list of factors, properties, aspects, components, criteria, tasks, or dimensions, the presence or amount of which are to be separately considered, in order to perform a certain task”. Unlike, for example, lessons learned, which tend to be descriptive, checklists by definition are prescriptive. That is, checklists constitute ‘actionable’ knowledge (Kokkonen 2006). “Checklists [guide] a user and act as verification after completion of a task, without necessarily leading users to a specific conclusion” (Ćatić and Malmqvist 2013). According to (Scriven 2000), checklists have the following benefits: help to reduce errors of omission; are relatively easy to understand and validate; reduce human biases (‘halo effect’ and ‘Rorschach effect’); reduce the problem of double weighting in evaluative tasks; and help to capture and transfer knowledge.

The primary objectives of using checklists include error reduction or best practice adherence, standing anywhere in-between an informal cognitive aid (memory recall) and a protocol (standardization and regulation of processes or methodologies) (Hales and Pronovost 2006). (Scriven 2000) distinguished between five types of checklists: Arbitrary (e.g., simple shopping checklist), sequential, weakly sequential (for psychological or efficiency reasons), partly or entirely iterative (e.g., problem-solving flowcharts) and diagnostic (e.g., decision-trees) checklists. Different types of checklists support certain levels of human performance and problem-solving. In the following, the examples of instantiation of checklists in aviation and product development are briefly reviewed.

Checklists in Aviation

In aviation, checklists under most circumstances are considered a mandatory part of the practice. Checklists are the flight protocols that all pilots are required to use before, during, and after flights. Completing a checklist from memory is considered a violation or error (Helmreich 2000). The two categories of checklists used in the cockpit include (Clay-Williams and Colligan 2015): normal and non-normal (or emergency) checklists.

Normal checklists are used as part of a regular flight practice to ensure that all necessary has been done (Degani and Wiener 1993), especially when the list of tasks is long to be accessed from memory and tasks are subject to interruptions. The typical normal checklists include preflight, cockpit, starting engine, landing, and shutdown checks (Hales and Pronovost 2006). Normal checklists have two types of execution strategies (Degani and Wiener 1993): (1) Do-verify, where the task is performed from memory first and then verified against the checklist (used in contexts with limited time); and (2) Call-Do-Respond, where tasks are divided between two or three pilots for calling, performing or verifying procedures. Both methods include at least action and verification steps (Hales and Pronovost 2006).

Non-normal (emergency) checklists, not part of the normal flight protocol, are used to guide the correction of error situations and may include checks for ground operation emergencies, take-off emergencies, landing emergencies, and fuel system failures (Hales and Pronovost 2006). Emergency checklists may contain boldface, non-

boldface or flowchart items, selected by the aircraft manufacturer depending on the likely severity of the problem and time available to solve it (Clay-Williams and Colligan 2015). Boldface items require immediate action, often executed from memory. Non-boldface checklists are used when the time is not critical. Checklists may also be instantiated as a flow chart or decision tree (Clay-Williams and Colligan 2015).

Furthermore, checklists for airplane maintenance and pilots' physical, mental, and emotional status evaluation before a flight have been developed (Hales and Pronovost 2006). In recent decades, aircraft manufacturers have transitioned from paper-based to electronic checklist systems to guide pilots through both normal and emergency procedures. Electronic checklists have helped to reduce errors further when compared to the paper-based checklists (Boorman 2001).

Checklists in Product Development

In product development, errors cause cost and time problems and poor quality of products. Moreover, errors can cause safety issues for producers as well as for product users. Standardized procedures and checklists as a common strategy have been used to manage errors (Hales and Pronovost 2006). As Masaki Imai stated in his seminal work, there can be no *kaizen* (continuous improvement) without standardization (Imai 1996). Checklists can be used as the means to implement and improve standards. In the following, as Toyota has been considered one of the leading companies in process standardization and implementation of checklists, the focus will be mainly on the Toyota product development system.

(Ward et al. 1995) argued that the second Toyota paradox, the first paradox being the Toyota Production System, is the set-based concurrent engineering: “[...] delaying decisions, communicating ambiguously, and pursuing an excessive number of prototypes, enables Toyota to design better cars faster and cheaper”. (Sobek and Liker 1998) described six organizational and managerial mechanisms underlying the second Toyota's paradox, including mutual adjustment, close supervision and integrative leadership as the social processes, and the standard skills, standard work practices and design standards as the means of standardization. Authors emphasize that these mechanisms are only useful when applied together.

Contrary to the US car manufacturers, Toyota has successfully standardized much of its development process (Sobek and Liker 1998). They have achieved it by carefully balancing the standardization of simplified work plans (“often fit on a single sheet”) and the flexibility of the process concept implementation in each vehicle program. The simplicity and flexibility help to develop “common understanding, and continuous improvement, while hard deadlines keep the project on track” (Sobek and Liker 1998). The product and standard development processes are always considered and designed together. Standard work plans are developed, implemented and updated by the designers and departments that use them.

At Toyota, engineering checklists (“lessons learned books”) are used by each participant to identify and record feasible design regions based on the current capabilities of the organization (Ward et al. 1995). Engineering checklists are highly visual and part or process specific for transferring experiences between vehicle programs (Morgan and Liker

2006). Engineering checklists comprise detailed information related to the aspects of, for example, functionality, manufacturability, government regulations, and reliability (Sobek and Liker 1998). Checklists also contain items on what can be done or not in an economic sense or items for incorporating new technologies for automation, cost reduction, quality improvement and so on (Morgan and Liker 2006). Checklists are used throughout the design process and particularly for design reviews. At the beginning of a new vehicle program, engineering teams exchange checklists to update each other on what is possible or not. In this way, assumptions are to be avoided.

According to (Sobek and Liker 1998), team members come to the review meetings with a prepared checklist of items they need to verify, and identified discrepancies between the checklists and designs become the points of discussion. When something new based on experience, analysis, experimentation, and testing is learned, it is added to the checklist. As such, checklists are continuously updated and become the means to explicate and transfer the accumulating knowledge of product development. The constant revision and updating of checklists, as part of the designers' work, also helps to develop a sense of ownership (Morgan and Liker 2006).

At Toyota, checklists are perceived to have the following benefits (Sobek and Liker 1998): improve face-to-face meetings, add predictability, facilitate organizational learning across vehicle programs, and make the knowledge to reside in the organization. Similar benefits have been recognized in other related domains, including the software engineering industry and engineering design in general.

In software engineering, although checklists have been mainly used for the inception processes of identifying defects and requirements engineering, (Kokkonen 2002) argued that checklists could also be used as part of the organizational memory system. Checklists can function as an experience-knowledge collection, experience-knowledge transferring and software process development tools (Kokkonen 2006). Similarly, according to (Firesmith 2005), checklists can support the software teams to make the state of the art the state of the practice by actually implementing the best known methods, techniques and knowledge.

The design of complex artefacts requires intensive knowledge-based activities to be carried out. For example, the questions-based approach has been proposed in engineering design literature. (Ahmed and Wallace 2001) proposed that questions-based design support system can aid novice designers to understand what they need to know in any given design context. (Grebici et al. 2009) developed five sets of generic questions that could guide the designers' inquiry into the subject matter. (Winkelmann and Hacker 2010) demonstrated that the use of interrogative questions stimulated reflection on solutions, which led to a significant improvement of the final solutions.

Common Characteristics of Checklists

Based on the literature review, checklists have been developed, implemented and maintained from three different perspectives: error reduction (Gawande 2010), process/performance (production) improvement (Ahmed and Wallace 2001; Grebici et al. 2009; Sobek and Liker 1998; Winkelmann and Hacker 2010) and knowledge management (Kokkonen 2006; Sobek and Liker 1998).

Benefits across the industries addressed above, include the error and human bias reduction (i.e., slips, lapses and mistakes); increased awareness of issues/aspects (e.g., safety issues); increased quality of services and products; improved team cohesion, communication and coordination; safer use of equipment and instruments; support for organizational learning.

The most elementary function of all checklists is that they are mnemonic devices (i.e., memory aids). Other functions of checklists depend on the use situation. In aviation, checklists for normal and non-normal (emergencies) situations have been developed. In the normal situation, the primary function of checklists is to avoid errors of omission and assure a best practice adherence (important for continuous improvement). In emergencies, checklists are used to support situation diagnosis and problem-solving. In the product development, checklists have been used to facilitate communication, coordination and team performance; to support quality improvement; and to provide the organizational memory system.

Five types of checklists for different use situations and functions have been proposed in the literature, including the arbitrary, sequential, weakly sequential (for psychological or efficiency reasons), partly or entirely iterative (e.g., problem-solving flowcharts) and diagnostic (e.g., decision-trees) checklists. Two common execution strategies include the “call-do-respond” and “do-verify”. Checklists are often paper-based, but electronic checklists are used as well (Hales and Pronovost 2006).

Different studies have emphasized that checklists are complex interventions and require careful design, development, and updating. These studies have argued that the most challenging aspect of implementing checklists is to convince the users to implement and maintain checklists (Hales and Pronovost 2006). (Degani and Wiener 1993) argued that it is also important to consider human factors design principles for designing and implementing checklists.

DISCUSSION

Design is a complex human activity subject to the opportunity of errors, resulting in costly mistakes. However, instead of focusing on individual mistakes, the focus should be on the system level, including in addition to the individual also the context. Joint cognitive systems approach describes human performance as the product of multi-layered activity. Human performance control involves multiple concurrent phases and modes of control. Different levels of performance are associated with different types of error, including slip, lapse, mistake, and violation.

Cognitive systems engineering seems useful for developing interventions for incorporating checklists into the daily work of designers. Different modes of human performance control, extending from reflective (interpretative) to reactive modes of action control, require the use of different types of checklists. Nowadays, as is the case in the aviation industry, it is also common to integrate the checklists into the artefacts for aiding human operators.

Checklists could also be conceptualized from the three views of production (management) proposed by (Koskela 2000). Checklists encoding standardized work procedures can help to reduce process variation, thus, waste (e.g., the seven preconditions

for the task and the seven wastes can be considered as checklists). Furthermore, checklists, when used for error reduction can help to improve quality, thus, value to the customer. Checklists as simple ‘to-do’ lists can define the necessary tasks to design an artefact, or part of the artefact

Before discussing the possibilities for developing support systems, it must be noted that human error management has roots in quality management (Shewhart 1931), although the term “human error” was not explicitly used there. Specifically, the approaches proposed by (Reason 1990) and Hollnagel and Amalberti (2001) overlap with ideas advocated in the quality management domain. Indeed, some of the main components of the Toyota error management include, for example, the mistake-proofing (‘poka-yoke’) and visual management (Shingo 1986; Ward et al. 1995).

DESIGN PROCESS SUPPORT SYSTEMS

The need for a systematic error and performance management has also been recognized in the building design context (Lopez et al. 2010). Inspired by the use of (electronic) checklists in the aviation industry, in this section, the main concepts relevant for developing a design (process) support system are addressed. The proposal relies on the premise that building designers use building information modelling (BIM).

DESIGN SUPPORT SYSTEM OR DESIGN PROCESS SUPPORT SYSTEM

(Ullman 2002) proposed the idea of ‘ideal mechanical engineering design support system’. However, the focus was on the artefact, the inner conceptualization of technical parts (functions and structures) of the design application. However, the performance of building designer is the product of interactions between the designer and the environment, including artefacts one uses to perform goal-oriented tasks.

Instead of focusing on the principles of product and their implementation in the product design applications, the focus should be, not neglecting the product view, on the process of design. That is, the design is primarily concerned with interactions involving human(s), object(s) and contexts together with the general aim of bringing about changes that are enabled and mediated by the situated subject (addresses interpretation) and object oriented (addresses causality) activities (Pikas 2019).

CONSTITUENTS OF DESIGN PROCESS SUPPORT SYSTEMS

In the domain of building design, the main objects include the context(s) (problem domain (global and local) and solution domain), humans (clients, users, and designers), and objects (product, BIM and checklists). These are embodied in the design process, which itself contains the following elements (Pikas 2019): modes of activity (analysis and synthesis), the categories of design activities (subject- and object-oriented), stages and phases, causal structure of design transformations (problem and solution state changes), iterations, and mental and external activities. This means that the contexts, humans, and objects of design are brought together through the design process.

DESIGN PROCESS MONITORING

The effective control of the design process requires close monitoring of the design process and providing information back to designers and design managers. But there are limitations with the typical approaches (Yarmohammadi and Castro-Lacouture 2018). Often, the daily process monitoring is left to the sole responsibility of individual designers, which may lead to conflicting activities and decisions. In the managerial context, due to the manual processes of collecting data, there is a considerable monitoring lag between the actual design process and feedback (Pikas 2019).

For the design process support system, a real-time design process monitoring system needs to be developed (e.g., a plugin for Revit could be developed). Although the focus in this study was on the measurement of outputs (model elements), and not taking into account contextual matters and designers activities, the feasibility of this has been already studied (Yarmohammadi and Castro-Lacouture 2018). The real-time monitoring of the design process could be then used to integrate the checklists into the daily work of designers.

CHECKLISTS FOR DESIGN PROCESS SUPPORT SYSTEMS

Checklists are not a new invention in the building design context. However, checklists have been used relatively narrowly, mostly for design reviews and inspections; e.g., the design review checklists of the Whole Building Design Guide (Prowler and Vierra 2012) or (Sannwald 2009). Furthermore, clash detection, say by Solibri, and automated code checking, are in a way automated checklists (Sacks et al. 2018). These types of checklists are primarily focused on the static aspects of design, i.e., design outputs.

Also, the process needs to be taken into account, and checklists for these need to be developed, be they manual or computerized. Furthermore, the two issues of developing checklists for building design include the development of the substance (contents) of checklists, and developing the media (channels) to deliver them to the point of use. The context-aware design process support system would help to automate the detection of a relevant checklist, and relevant item on that list, and assure compliance to the standards.

CONCLUSION

An initial framing for design process support systems was developed in this study. The purpose of a design process support system would be to facilitate the error, performance and knowledge management; needed because design as a complex activity is prone to errors. Cognitive engineering systems together with the process perspective of design can facilitate the theoretical development framework for design process support systems. If the processes of BIM-enabled design can be monitored, then checklists could be incorporated into the design process to standardize work, facilitate communication and coordination, improve quality, and enable knowledge transfer between projects.

However, as this is only an initial framing, many aspects of the design process support systems still need to be addressed. For example, in this work, there was no room to specifically address the content and relationships between the different constituents of design process support systems. Also, how exactly the monitoring of designers work can be done and what kind of analytics (probably something along the lines of process mining)

is required to make sense of the raw data. Similarly, the kinds of visualizations needed to communicate the checklists in real time to designers at work are also significant. All these are important questions and need further study.

REFERENCES

- Ahmed, S., and Wallace, K. M. "Developing a support system for novice designers in industry." *Proc., International Conference on Engineering Design (ICED 01), Design Management–Process Information, IMechE*, 75-82.
- Bedny, G., and Meister, D. (2014). *The Russian theory of activity: Current applications to design and learning*, Psychology Press.
- Behm, M. (2005). "Linking construction fatalities to the design for construction safety concept." *Safety science*, 43(8), 589-611.
- Boorman, D. (2001). "Today's electronic checklists reduce likelihood of crew errors and help prevent mishaps." *ICAO Journal*.
- Ćatić, A., and Malmqvist, J. (2013). "Effective method for creating engineering checklists." *Journal of engineering design*, 24(6), 453-475.
- Clay-Williams, R., and Colligan, L. (2015). "Back to basics: checklists in aviation and healthcare." *BMJ Quality & Safety*, 24, 428-431.
- Degani, A., and Wiener, E. L. (1993). "Cockpit checklists: Concepts, design, and use." *Human factors*, 35(2), 345-359.
- Eurostat (2018). "Accidents at work statistics - Fatal accidents at work, 2015." DG Employment and Social Affairs – Eurostat., URL: https://ec.europa.eu/eurostat/statistics-explained/index.php/Accidents_at_work_statistics, Access date: 05.02.2019.
- Firesmith, D. (2005). "Quality Requirements Checklist." *Journal of Object Technology*, 4(9), 31-38.
- Gawande, A. (2010). *Checklist manifesto*, Penguin Books India.
- Gorovitz, S., and MacIntyre, A. (1975). "Toward a theory of medical fallibility." *Hastings Center Report*, 5(6), 13-23.
- Grebici, K., Aurisicchio, M., and Bracewell, R. "Guiding engineering design activities through a question based approach." *Proc., ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, 1449-1459.
- Hales, B. M., and Pronovost, P. J. (2006). "The checklist—a tool for error management and performance improvement." *Journal of critical care*, 21(3), 231-235.
- Helmreich, R. L. (2000). "On error management: lessons from aviation." *Bmj*, 320(7237), 781-785.
- Hollnagel, E. (2002). "Cognition as control: A pragmatic approach to the modelling of joint cognitive systems." *IEEE Journal of Systems, Man and Cybernetics*.
- Hollnagel, E., and Amalberti, R. (2001). "The emperor's new clothes: Or whatever happened to "human error"." *Proceedings of the 4th international workshop on human error, safety and systems development*, Linköping University, 1-18.

- Hollnagel, E., and Woods, D. (1999). "Cognitive systems engineering: new wine in new bottles." *International Journal of Human-Computer Studies*, 51, 339-356.
- Hollnagel, E., and Woods, D. D. (2005). *Joint cognitive systems: Foundations of cognitive systems engineering*, CRC Press.
- Imai, M. (1996). "KAIZEN: the key to Japan's competitive success. Editorial CECSA, Mexico." *Google Scholar*.
- Klein, G., Ross, K. G., Moon, B. M., Klein, D. E., Hoffman, R. R., and Hollnagel, E. (2003). "Macro cognition." *IEEE intelligent systems*, 18(3), 81-85.
- Kokkonen, J. (2002). "Checklists as a tool for collecting experience knowledge." *Proceedings of IASTED International Conference of Information and Knowledge Sharing (IKS 2002)*, St. Thomas, Virgin Islands, USA, 223-228.
- Kokkonen, J. (2006). "Experiences from generating checklists." *Proceedings of the IASTED International Conference on Knowledge Sharing and Collaborative Engineering (KSCE2006)*, St. Thomas, Virgin Islands, USA, 51-56.
- Kokkonen, J. (2006). "A preliminary model for generating experience knowledge based artifacts." *Proceedings of the 39th Hawaii International Conference on System Sciences (HICSS'06)*, IEEE, Hawaii, USA, pp. 10.
- Koskela, L. (2000). *An exploration towards a production theory and its application to construction*, VTT Technical Research Centre of Finland.
- Lindemann, U., Maurer, M., and Braun, T. (2009). "Complexity in the context of product design." *Structural Complexity Management: An Approach for the Field of Product Design*, Springer Berlin Heidelberg, Berlin, Heidelberg, 21-42.
- Lopez, R., Love, P. E., Edwards, D. J., and Davis, P. R. (2010). "Design error classification, causation, and prevention in construction engineering." *Journal of Performance of Constructed Facilities*, 24(4), p. 399-408.
- Love, T. (2002). "Constructing a coherent cross-disciplinary body of theory about designing and designs: some philosophical issues." *Design Studies*, 23(3), 345-361.
- Morgan, J. M., and Liker, J. K. (2006). "The Toyota product development system." *New York, US*.
- Pikas, E. (2019). "Causality and Interpretation: Integrating the Technical and Social Aspects of Design." Doctor of Science (Technology), Aalto University, Espoo, Finland.
- Prowler, D., and Vierra, S. (2012). "Whole building design." *Whole Building Design Guide*.
- Reason, J. (1990). *Human error*, Cambridge University Press, Cambridge, UK.
- Reason, J. (2000). "Human error: models and management." *Bmj*, 320(7237), 768-770.
- Sacks, R., Eastman, C., Lee, G., and Teicholz, P. (2018). *BIM Handbook: A Guide to Building Information Modeling for Owners, Designers, Engineers, Contractors, and Facility Managers*, Wiley.
- Sannwald, W. W. (2009). *Checklist of library building design considerations*, American Library Association.
- Saurin, T. A., Formoso, C. T., and Cambraia, F. B. (2008). "An analysis of construction safety best practices from a cognitive systems engineering perspective." *Safety science*, 46(8), 1169-1183.

- Scriven, M. (2000). "The logic and methodology of checklists." Retrieved from: http://www.wmich.edu/evalctr/archive_checklists/methodology_dec07.pdf.
- Senders, J. W., and Moray, N. P. (1991). *Human Error: Cause, Prediction, and Reduction*, Taylor & Francis.
- Shewhart, W. A. (1931). *Economic control of quality of manufactured product*, ASQ Quality Press.
- Shingo, S. (1986). *Zero quality control: Source inspection and the poka-yoke system*, CRC Press.
- Sobek, I., and Liker, J. K. (1998). "Another look at how Toyota integrates product development." *Harvard business review*, 76(4), 36-47.
- Suchman, L. (1987). "Plans and situated action." Cambridge: Cambridge University Press.
- Ullman, D. G. (2002). "Toward the ideal mechanical engineering design support system." *Research in Engineering Design*, 13(2), 55-64.
- Wachs, P., and Saurin, T. A. (2018). "Modelling interactions between procedures and resilience skills." *Applied ergonomics*, 68, 328-337.
- Ward, A., Liker, J. K., Cristiano, J. J., and Sobek, D. K. (1995). "The second Toyota paradox: How delaying decisions can make better cars faster." *Sloan management review*, 36, 43-43.
- Winkelmann, C., and Hacker, W. (2010). "Question-answering-technique to support freshman and senior engineers in processes of engineering design." *International Journal of Technology and Design Education*, 20(3), 305-315.
- Woods, D. "Rasmussen's SRK 30 Years Later: Is Human Factors Best in 3's?" *Proc., Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, SAGE Publications Sage CA: Los Angeles, CA, 217-221.
- Woods, D. D., and Hollnagel, E. (2006). *Joint cognitive systems: Patterns in cognitive systems engineering*, CRC Press.
- Woods, D. D., Johannesen, L. J., Cook, R. I., and Sarter, N. B. (1994). "Behind human error: Cognitive systems, computers and hindsight." Dayton University Research Institute.
- Yarmohammadi, S., and Castro-Lacouture, D. (2018). "Automated performance measurement for 3D building modeling decisions." *Automation in Construction*, 93, 91-111

