Teaching Software Engineering with LEGO Serious Play

Stan Kurkovsky
Central Connecticut State University
kurkovsky@ccsu.edu

ABSTRACT

LEGO Serious Play (LSP) is a methodology that helps people brainstorm and discuss complex ideas through storytelling and metaphors. LSP has been successfully applied in higher education as a mechanism for team building and promoting creativity. In this paper, we discuss using LSP to teach several core software engineering topics through hands-on case studies. Initial results suggest that LSP has a positive impact on student learning, while also improving student engagement with the course material. This paper describes the details of two LSP-based case studies along with many practical aspects of using LSP to teach software engineering.

Categories and Subject Descriptors

D.2.0 [**Software Engineering**]: General.
K.3.2 [**Computing Milieux**]: Computer and Information Science Education – *Computer science education.*

General Terms

Management, Design, Human Factors.

Keywords

Software Engineering, case studies, LEGO Serious Play.

1. INTRODUCTION

Software engineering courses often serve as an integrative experience where students apply programming skills together with their knowledge of many Computer Science areas. However, studying software engineering must be more than just participating in a capstone course where existing knowledge and skills are put to practice. There are many important principles and concepts that are central to the practice of modern software engineering, which usually are not covered in other courses forming the traditional computing curriculum. These include requirements engineering, emergent properties, socio-technical systems, etc. Given the engineering nature of the discipline, one of the best ways to learn these principles is usually to apply them in a practical context, such as a course project or a case study.

This paper discusses the application of LEGO Serious Play (LSP) to studying software engineering concepts through hands-on case studies. While there are no published reports concerning the use of LSP in software engineering education (except as a team-building method), LEGO bricks are being successfully used to introduce agile software methods [11,18]. Most notably, LEGO Mindstorms have become a popular learning tool in engineering and computing classrooms and provided a strong positive impact on student learning over the last 15 years [5]. LSP is a highly participative facilitated methodology that helps people brainstorm and discuss complex ideas through storytelling that revolves around the metaphors represented through LEGO models. In LSP, LEGO models only act as a starting point for discussion. Students open up and start brainstorming ideas together, explore and often find unexpected solutions to the presented problems. LSP has a very strong theoretical background rooted in constructionist learning theory, as described in Section 2. Many LSP practitioners refer to this methodology as a 'programming language,' using which it is possible to build a solution for the given problem. Practical rules of LSP described in Section 3 are based on the 'build-share-reflect' sequence, where the facilitator poses a question or a challenge related to the problem at hand, in response to which the participants build LEGO models, share stories about them, and reflect on their understanding of the problem.

The author has developed and tested a number of LSP-based case studies to teach software engineering concepts, such as architectural patterns, requirements engineering, and software dependability. Section 4 lays out this approach, which leverages intentional emergence to combine the fragmented knowledge of each individual student in a way that would help the team gain a deeper and fuller understanding of the given concept. The effects of these case studies on student learning are discussed in Section 5. Section 6 describes the lessons learned and practical implications of using LSP in the classroom.

2. SERIOUS PLAY AND LEGO

Rieber et al [20] define *serious play* as "an intensive and voluntary learning interaction consisting of both cognitive and physical elements. Serious play is purposeful, or goal oriented, with the person able to modify goals as desired or needed. Most importantly, the individual views the experience of serious play as satisfying and rewarding in and of itself and considers the play experience as important as any outcomes that are produced as a result of it." Serious play is an example of an optimal life experience or *flow* proposed by Csíkszentmihályi [4]. Flow is a state of concentration when the person is completely immersed and is "carried by the flow" of the activity, often ignoring the passage of time. The key aspect of serious play is in experiencing flow due to the satisfaction of understanding something complex, confusing, or previously unknown.

Theoretical foundations of LEGO Serious Play were developed by Roos and Victor as a novel methodology to harness creativity and imagination for business strategy development [22]. This approach is grounded in Jean Piaget's *constructivism* theory stating that children develop their knowledge based on their hands-on experiences of the surrounding world. Seymour Papert later extended Piaget's theory. Papert's concept of *constructionism* rests on the idea that learning occurs when we build something external to ourselves that is related to the studied subject. One might say that constructivism explains how we build knowledge in our heads, while constructionism explains how creating tangible objects solidifies that knowledge. Constructivism and constructionism helped us move away from

ITICSE '15, July 04 - 08, 2015, Vilnius, Lithuania
Copyright 2015 ACM 978-1-4503-3440-2/15/07…$15.00
http://dx.doi.org/10.1145/2729094.2742604

traditional lectures to actively engage students in hands-on experiences that help them make sense of the real world.

The central idea behind the LSP methodology is that "when you build in the world, you build in your mind" [19]. This refers to our own mental models that we use to make sense of the world, whether they represent our work environment, computer programming concepts, or personal beliefs. Using one's hands is the key concept of constructionism. Official LSP training manual suggests that touching, manipulating, and constructing physical objects with our hands activates a richer kind of learning. Extending the constructionist concept of "thinking with object," LSP can be viewed as a language for articulating complex and tacit knowledge [9]. Instead of LEGO bricks, one could use modeling clay, construction paper, sticky notes, or some other kind of medium. Many serious play practitioners agree, however, that LEGO bricks are the preferred choice because they are much easier to work with, require no special skills, or a cleanup after playing. It is important to point out that the richness of LSP is not in the medium used to construct models, but rather in what these models represent [15]. The modularity of LEGO bricks allows the participants to continuously modify their models and elaborate on their stories. At the same time, participants are usually quick to understand that none of their models will look very realistic and will have little meaning without providing their own stories. This observation helps the participants stop worrying about their artistic abilities and encourages them to build models with rich metaphorical meaning [6].

LSP uses special sets of bricks that are designed to "inspire the use of metaphors and story-making" [19], such as minifigures, animals, money, etc. The facilitator poses a question or a challenge, e.g. "Build a model of a nightmare professor," in response to which participants build their individual models and then explain their model and its meaning to everyone. The etiquette of LSP ensures that everyone gets to express their ideas without being influenced by others. When a question is posed, everyone starts with building their models, and only after that the story-telling begins. This approach significantly helps students who otherwise might be shy and hesitant to engage in a classroom discussion. At the same time, this helps to quiet down those students who are always eager to jump into a discussion and/or have an answer to every question. This democratic process and the level playing field created by the universal language of LEGO bricks not only gives everyone an equal opportunity to participate, but also creates a playful and positive shared experience of discussing and making sense of a complex problem.

3. THE PRACTICE OF LSP

LSP has been developed for use in facilitator-led workshops consisting of 6-10 participants. Workshops can range in duration from 1.5 hours to two days. During each workshop, participants build a series of models with the goal of team building, gaining a deeper understanding of a complex problem, or developing a strategy. The process of building a model includes four steps:

1. Facilitator poses a question/challenge;
2. Participants build models;
3. Participants explain their models by sharing stories; and
4. Participants reflect on their understanding of the models and their meanings.

There are several fundamental ethical principles that guide LSP and help each workshop stay focused and productive. Each participant owns his or her model and only that participant can change it or give it a meaning. All discussions are focused on the models only and not on their owners.

Each LSP workshop begins with a skill-building exercise aimed to stimulate different types of imagination described in [22]. This warm-up exercise guides the participants through basic LEGO construction skills, building representations and metaphors, and explaining them through story telling. Upon completing this exercise, participants should be comfortable enough with LSP to begin working on the tasks directly related to the specific objectives of the workshop.

LSP methodology defines seven Application Techniques [10]:

AT-1. Building individual models and stories;
AT-2. Building shared models and stories;
AT-3. Creating a landscape;
AT-4. Making connections;
AT-5. Building a system;
AT-6. Playing emergence and decisions; and
AT-7. Extracting simple guiding principles.

Depending on the goals and the context, an LSP workshop can include any combination of the application techniques, but it always starts with building individual models and stories (AT-1). We illustrate this principle in detail in Sections 4.2 and 4.3.

4. LSP AND SOFTWARE ENGINEERING

The value of serious play in education has long been recognized [21]. Despite the fact that LSP is fundamentally based on several educational theories, there are just a handful of reports on using LSP in the context of education. LSP has been applied in a few higher education areas: reflecting on their learning process by creative arts students [7], fostering creativity of students studying management information systems [17], team building in a graduate information technology program [23] and engineering [1], developing assessment strategies by students in a post-secondary teacher education program [16], and leadership development in an industrial engineering program [8]. The majority of these reports do not indicate that LSP was used to teach students the 'core' topics of the corresponding courses or programs. On the contrary, our approach uses LSP to create a series of hands-on case studies that help students master the major topics in a software engineering course along with building better teamwork skills and promoting creativity.

LSP has a proven track of success in professional software engineering. Requirements elicitation in human-computer interaction has been studied [2] and later formulated as a formal LSP-based technique called "User Requirements with LEGO" (URL) [3]. As a custom designed workshop focusing on web-based projects, URL guides the participants through identifying their roles, types and expectations of the users, modeling the content and key functionality of the website, connecting stakeholders with website content, features, and users, and identifying any possible shortcomings (e.g. website functionality that no users need, or a feature demanded by a large population of users that does not have an adequate level of support).

Teamwork is central to software engineering. Effective teams need to possess an ability to communicate effectively, develop shared mental models, and remain motivated while working together to achieve the project objectives [13]. The concept of a *shared mental model* refers to the team's mutual understanding of their tasks and objectives, the workflow process, and their teamwork strategy to reach the goals [12]. Although originally developed as a strategy-making tool, LSP is also an effective team-building methodology, which encourages participants to share their assumptions and ideas [1,24]. Using LSP can help the team build a shared mental model through continuously expressing individual ideas by building and explaining the models to the team. As illustrated in the case studies below, shared models (AT-2) help the team build a consensus by integrating the ideas of each team member, while landscape models (AT-3) enable forming a shared understanding constructed out of individual and sometimes different perspectives.

Mabogunje et al describe the phenomenon of *intentional emergence* in the context of product development process [14]. Emergence may be viewed as the ability of a complex system to produce behaviors or properties as a result of interactions among its components, which by themselves cannot produce such behaviors or properties. LSP is used as a mechanism that can make emergence intentional. Our approach leverages intentional emergence by combining individual and often partial knowledge of a certain software engineering concept and the background of each student in a way that would help the team gain a deeper and fuller understanding of that concept. We use LSP in a number of hands-on case studies that are aimed to reinforce student understanding of a particular software engineering concept, such as requirements engineering, software design patterns, or software dependability. A lecture on the given topic precedes the case study. Each participating student brings in their current understanding of the topic supplemented by any relevant experience with it they may have accumulated from other courses or practical experience. During an LSP-based case study, students build models and tell stories explaining their views and understanding of the topic. The overarching goal of all LSP-based case studies is to leverage the combined knowledge, experience, and backgrounds of all participating students. Throughout the case study, students are guided to form a shared understanding of the given concept that combines all correct and relevant elements of the topic in the focus of the case study, which would ultimately reinforce and deepen their mastery of the course material.

4.1 LSP-based Case Studies in the Classroom

Below we describe the experience of using LSP in a senior software engineering course with 20 students, which met twice a week for 75 minutes. A typical LSP experience begins with the skill-building exercise, to which the author dedicated an entire 75-minute block.

The introductory skill-building exercise consisted of the following three challenges. The *building challenge* requires student to build a simple structure (a tower) in order to (re)acquaint students with the LEGO bricks. No two resulting structures are ever the same, which serves as a learning point: we all have our own perspectives and our models are unique. The *metaphor challenge* requires students to build a model from Imaginopedia, a brochure included with the LSP Starter Kit. The students are then asked to modify their creations, if necessary, so that the model would illustrate the concept of being a CS student, a software engineer, or an IT professional. Then, students were asked to explain the meaning of their models. This challenge helps students see their models in an entirely different way: not as a scale model of a real-world object, but as a metaphor that can be used to tell a story. In the *story-making challenge* students were asked to build a model from scratch that would represent a nightmare assignment, a project, or a ‘professor from hell,’ and then tell a story explaining their model. Expert LSP facilitators agree that it is always easier to begin by building models illustrating some extreme qualities of something that is very familiar to all participants. During this challenge, students are encouraged to ask each other to explain specific details of their models, which eases them to focus on their models while communicating the elements of complex concepts.

For all subsequent case studies, the class was split into two groups that met in separate classrooms and worked on the same assignment: one group using LSP and building LEGO models, and the other group using a whiteboard as a medium to support their discussion. The use of LSP alternated between the groups so that all students would receive a comparable exposure to this methodology. All students were graded based on the worksheets that they completed while working on a case study. Each student working in the LSP group used a single LSP Starter Kit, which costs about $37.

All LSP-based case studies went beyond building individual models and included building either a shared or a landscape model, both of which promote team building and creating of shared understanding. These two kinds of models force students to compare their thoughts and views on the same concept, which helps each student correct any possible misconceptions and crystallize their understanding of that concept.

The author piloted a total of five LSP-based case studies:

1. Requirements engineering: formulate and refine use cases of a software system;
2. Software architecture: identify architectural components of a system and choose a suitable architectural style;
3. Design patterns: choose a software design pattern best suited to implement a software component;
4. Socio-technical systems: study emergent properties and behaviors in a complex socio-technical system; and
5. Dimensions of dependability: identify, analyze, and mitigate the risks to dependability of a complex system.

Each case study is structured as a typical LSP workshop, in which instructor poses a challenge relevant to the corresponding software engineering topic, and students build and explain their models. Students reflected on their understanding of that topic both through discussions and by completing a worksheet, which was later graded using a rubric consisting of 8-10 criteria.

4.2 Case Study: Requirements Engineering

The objective of this case study was to identify, analyze, and refine the use cases of a software system. This case study was about a hypothetical software system called Programming Assignment Submission System (PASS). The objective of PASS is to help multiple CS instructors by automating the process of managing programming assignments in their courses. The core functionality of PASS is described as follows: for instructors – post the assignments and suggested solutions, review and grade the work submitted by students; for students – submit their solutions, get email reminders when an assignment is almost due and when it is overdue; and for teaching assistants – view the assignments and submitted student work. Using their worksheets, students were required to draw a UML use case diagram for PASS paying a special attention to using <<*include*>> and <<*extend*>> relationships, and to write a description of one non-trivial use case using the table format, a sample of which was provided. The description of system functionality given to the students was intentionally somewhat ambiguous and open to interpretation in order to promote student discussion.

The room was arranged so that each student had an individual building table. Students were asked to place and discuss their models at a separate discussion table. Each student used one LSP Starter Kit.

The students were first asked to identify the actors in PASS and build their models. The models were brought to the discussion table and placed on small platforms so that all models representing the same actor would be on the same platform. The most obvious actors are a student, an instructor, and a teaching assistant, but other possibilities also included a system administrator and a secretary.

Then, each student was asked to build one individual model of a use case for PASS (AT-1). Each student built his or her own individual model and the discussion did not commence until after everyone was done building. This ensured that the reasoning of each student was not affected by others and that everybody's voice was heard. The models of use cases were then brought to the discussion table where each student told the story of how their use case plays out in PASS. Models of similar or duplicating use cases were grouped together and their owners were asked to build a single shared model upon which all of them must agree (AT-2). In order to do this, students identified the most crucial component (typically consisting of several bricks) in their individual model and then constructed a shared model out of these components. Most importantly, the owners of the shared model came up with a unified story explaining how their new model represented the corresponding element of PASS's functionality. This ensures that each student has some ownership of the use case on which they worked and that their views and opinions are taken into account. The process of building use case models can be repeated if one or more use cases have not yet been identified and if the time allows. A collection of shared use case models now represented a landscape model (AT-3), which portrayed different aspects of the system's functionality. When the previously built actors were connected with the models of use cases using the special LEGO elements, the result represented a connection model of PASS (AT-4), which by now resembled a UML diagram. Students were asked to discuss the role of each use case within the system so that any candidates for using the UML <<*extend*>> or <<*include*>> relationship could be identified and properly reconnected. Now that a complete model was built, students drew the resulting UML use case diagram on their worksheet and wrote a detailed description of one non-trivial use case of their choice. A subset of the resulting LEGO models and a fragment of the corresponding UML use case diagram are shown in Figure 1.

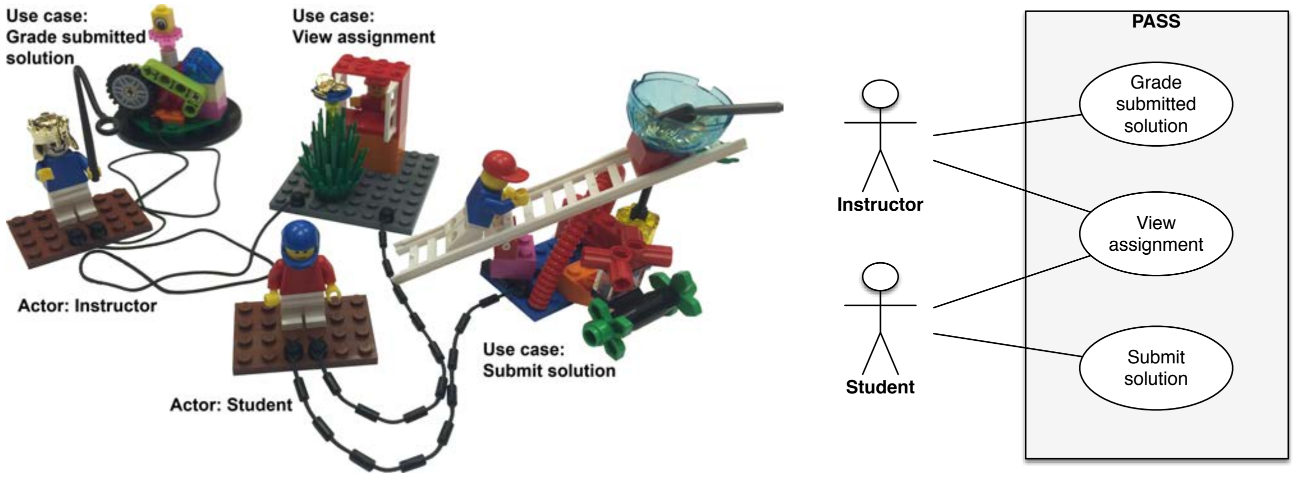

Figure 1. A subset of LEGO models of actors and use cases in PASS and the corresponding UML use case diagram.

Figure 2. LEGO models of four dimensions of system dependability.

4.3 Case Study: Dimensions of Dependability

The objective of this case study was to identify, analyze, and mitigate the risks to dependability of a complex socio-technical system. In particular, this case study explored four different dimensions of dependability: availability, reliability, safety, and security. It is important to note that system dependability also includes other properties, such as survivability, maintainability, integrity, etc. These four properties were examined in the context of four different systems: an automatic parking garage gate, a smartphone, a digital picture frame, and a traffic light control system. We used two decks of cards to create random combinations, e.g. reliability of a parking garage gate or security of a digital picture frame. Each card in one deck was labeled with one dimension of dependability, while the other deck included different systems. Each student picked a random card from each of the decks. Given these selections, students were asked to build a model and come up with an event/scenario illustrating the corresponding risk to dependability in the given system.

Once the individual models were built (AT-1), students explained their models grouped by the dependability feature, e.g. reliability. After all stories related to system reliability were shared, all corresponding models were grouped together into a landscape model (AT-3) and the students were asked to reflect upon them: does each model create a good scenario illustrating a risk to the system's reliability? Students were asked to pick the best or the most relevant scenario, improve upon it, if necessary, and briefly describe that scenario in their worksheets. This process was repeated for the models related to each of the remaining dimensions of dependability.

Once all models were discussed and the landscape models were built (Figure 2), students performed a risk-based analysis of their models, given the system/scenario combinations. Students were asked to identify and describe the specific risk factors, analyze and assess that risk based on its severity/probability, decompose the risk to identify its root cause, and reduce the risk by choosing an appropriate risk mitigation strategy to improve system dependability (risk avoidance, risk detection and removal, or risk tolerance). Students were asked to explain their reasoning to the team based on the models they've built and to write down the key points in their worksheets.

5. EFFECTS ON STUDENT LEARNING

During each case study, students completed a graded worksheet consisting of 8-10 questions gauging the level of student skills in the cognitive domain of Bloom's taxonomy, as shown in Table 1 (not all case studies included questions aimed at assessing evaluation skills, and none had questions related to the synthesis skills). During each case study, the test group used LSP, while the control group solved the same problem without using metaphors and LEGO models. Students in the test and control groups alternated to ensure a more equal exposure to the LSP techniques by the entire class. Individual student work in each case study was scored on the scale of 0 to 5. The sample size *n* indicates the number of questions testing the given skill level multiplied by the size of the student pool; there were 10 students in the test and 10 students in the control group. The difference between student grades in the test and control groups was assessed using a one-tailed t-test to determine the effect of using LSP on student learning. As shown in Table 1, students who used LSP attained a higher level of skills (statistically significant, $p<0.05$) in the areas of comprehension, application, and analysis. The level of evaluation skills was also higher for students who used LSP, but with a weaker statistical significance ($p<0.1$).

At the end of the semester, students were asked to reflect whether LSP-based case studies helped them learn the course material:

I believe that the concepts learned with LEGO go deep in my mind. I will remember them because I played. I explained my design to my teammates. When I build with LEGO, it creates more meaning to the concepts from each lecture.

I tend to think verbally and conceptually, but forcing me to visualize added an extra dimension to my learning.

I recall many of the LSP case studies better than the whiteboard sessions. The requirements engineering case study in particular was made very clear by constructing representations of system components, actors, etc.

Both types of case studies were useful, but using LEGO made it easier to understand the systems being discussed. It also made it easier to explain the details to other students because I could show them using my LEGO designs.

Table 1. Comparative analysis of student learning with LSP.

Characteristic	Group	Mean	SD	n	Δ	t	p
Comprehension	test	4.68	0.90	110	0.34	2.772	0.007
	control	4.35	1.09				
Application	test	4.66	0.69	150	0.17	2.006	0.047
	control	4.49	0.99				
Analysis	test	4.61	0.98	130	0.22	2.044	0.043
	control	4.39	1.04				
Evaluation	test	4.45	0.87	60	0.28	1.886	0.064
	control	4.17	1.14				

6. CHALLENGES AND LESSONS LEARNED

The author's experience shows that some faculty are apprehensive about using LEGO bricks in a university classroom. A common perception is that this is a just a toy and it cannot be used for anything other than developing spatial reasoning skills in first-year engineering students. Some can be persuaded by the global track record of LSP, but a few will always believe that it is just a corporate training technique and a waste of time in the classroom.

Timing considerations are extremely important when planning an LSP activity that is supposed to fit a single class period. All LSP case studies described here were designed for 75-minute blocks. Building and reflecting must be carefully timed because a properly designed serious play activity will put both students (and professors too!) in the state of flow when they can be easily carried away and keep on tinkering with their models and discussing them. It is sufficient to allow 3-5 minutes for building (nobody gets bored if they finish building quickly) and, based on the team size, 5-15 minutes for discussion/reflection.

Given the timing considerations, a detailed script of the case study must be developed beforehand. It should include full details of all building steps and all instructions that will be given to the students. Ideally, each script should be given a trial run to test whether the instructions are sufficient, the tasks have the appropriate level of difficulty, and the timing is realistic. The script should also include any other elements that must be planned in advance, e.g. table arrangement, additional materials, etc.

A short debriefing session is always useful at the end of every LSP activity to help reinforce the key concepts and learning points. This also gives students an opportunity to provide feedback about what went well and what did not. The author's experience shows that it is immensely important for the facilitator to write down their own reflection so that the activity can be improved the next time it is offered.

From the outset, a few students were hesitant about using LEGO in what they perceived as a very serious course. Most of them changed their mind during the LSP skill-building exercise, but a couple remained skeptical throughout the entire course. It is important to identify such students soon enough after introducing LEGO and LSP to attempt changing their minds.

Finally, it is crucial to point out the importance and necessity of professional LSP certification that provided the author with the vital background, insight, and skills for developing LSP activities.

7. CONCLUSIONS

The analysis of quantitative and qualitative data suggests that LSP had a positive impact on student learning in a senior software engineering course. Compared to a traditional method of running case studies, LSP-based activities resulted in students reaching higher skills in many levels of Bloom's taxonomy. Formal data along with written and informal feedback from students suggests that LSP helped improve soft skills, such as teamwork and oral communication. Students indicated that LSP increased their motivation, promoted creativity, and improved retention by actively engaging students into the coursework.

Focusing on LEGO models helped students open up and start discussing complex software engineering concepts with their team. Some students reported that having their models in front of them made them feel more relaxed, which, in turn, helped alleviate their fear of public speaking. Students very frequently took pictures of their models and shared them on social media networks, which not only helped them to extend their reflection of the models, but also indicated that they enjoyed LSP case studies and were proud of their work.

Running LSP activities is always fun and helps break up the routine. LSP helps foster a creative, playful, and imaginative classroom atmosphere; consequently, many students reported looking forward to more LSP-based case studies.

8. ACKNOWLEDGMENTS

The author is thankful to Robert Rasmussen, LSP's main architect, for showing and teaching the *serious* side of LEGO.

9. REFERENCES

[1] Bulmer, L. 2011. The Use of LEGO Serious Play in the Engineering Design Classroom. *Proceedings of the Canadian Engineering Education Association*.

[2] Cantoni, L., Botturi, L., Faré, M., and Bolchini, D. 2009. Playful Holistic Support to HCI Requirements Using LEGO Bricks. In *Human Centered Design*, 844-853. Springer.

[3] Cantoni, L., Faré, M., and Frick, E. 2011. *URL: User Requirements with Lego. Ver. 1.0*. May 2011. http://www.webatelier.net/url-user-requirements-with-lego

[4] Csíkszentmihályi, M. 1991. *Flow: The psychology of optimal experience*. Harper & Row.

[5] Danahy, E., Wang, E., Brockman, J., Carberry, A., Shapiro, B., and Rogers, C. B. 2014. LEGO-based Robotics in Higher Education: 15 Years of Student Creativity. *Int. J. Adv. Robot Syst.*, *11*, 27.

[6] Gauntlett, D. 2007. *Creative explorations: New approaches to identities and audiences*. Routledge.

[7] James, A. R. 2013. Lego Serious Play: a three-dimensional approach to learning development. *J. Learning Development in Higher Education*, (6).

[8] Jentsch, D., Riedel, R., and Mueller, E. 2013. Flow and Physical Objects in Experiential Learning for Industrial Engineering Education. In *Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services,* 566-573.

[9] Kristiansen, P., Hansen, P. H. K., and Nielsen, L. M. 2009. Articulation of tacit and complex knowledge. In *13th International Workshop of the IFIP WG 5.7 SIG,* 77-86.

[10] Kristiansen, P. and Rasmussen, R. 2014. *Building a Better Business Using the Lego Serious Play Method*. John Wiley & Sons.

[11] Krivitsky, A. 2011. *Scrum Simulation with LEGO Bricks*. http://www.lego4scrum.com.

[12] Landy, F. J. and Conte, J. M. 2010. *Work in the 21st century: An introduction to industrial and organizational psychology*. John Wiley & Sons.

[13] Leonard, H. S. and Freedman, A. M. 2000. From scientific management through fun and games to high performing teams: A historical perspective on consulting to team-based organizations. *Consulting Psychology Journal: Practice and Research*, *52*(1), 3-19.

[14] Mabogunje, A., Hansen, P. K., Eris, O. and Leifer, L. 2008. Product Design and Intentional Emergence facilitated by Serious Play. *Proc. of NordDesign 2008 Conf.*

[15] McCusker, S. 2014. Lego, Seriously: Thinking through building. *Intl. J. Knowledge, Innovation and Entrepreneurship, 2*(1), 27-37.

[16] Nerantzi, C. and Despard, C. S. 2014. Do LEGO Models Aid Reflection in Learning and Teaching Practice?. *J. Perspectives in Applied Academic Practice*, 2(2), 31-36.

[17] Oberer, B. 2013. Integrating Creative Learning Elements in Higher Education Shown in the Example of a Management Information Systems Courses. *Education*, 3(6), 319-324.

[18] Paasivaara, M., Heikkilä, V., Lassenius, C. and Toivola, T. 2014. Teaching students Scrum using LEGO blocks. In *Proc 36th Intl. Conf. on Software Engineering,* 382-391.

[19] Rasmussen, R. 2006. When You Build in the World, You Build in Your Mind. *Design Management Review*, 17(3), 56-63.

[20] Rieber, L. P. and Matzko, M. J. 2001. Serious design of serious play in physics. *Educational Technology*, 41(1), 14-24.

[21] Rieber, L. P., Smith, L. and Noah, D. 1998. The value of serious play. *Educational Technology*, 38(6), 29-37.

[22] Roos, J. and Victor, B. 1999. Towards a new model of strategy-making as serious play. *European Management J.*, *17*(4), 348-355.

[23] Scharlau, B.A. 2013. Games for teaching software development. In *Proc 18th ACM Conf. on Innovation and technology in computer science education*, 303-308.

[24] Schulz, K.-P., and Geithner, S. 2011. The development of shared understandings and innovation through metaphorical methods such as LEGO Serious Play. *Intl. Conf. on Organizational Learning, Knowledge and Capabilities*.