"Free" Innovation Environments: Lessons learned from the Software Factory Initiatives

Davide Taibi, Valentina Lenarduzzi
Free University of Bolzano-Bozen
Bolzano-Bozen, Italy
e-mail: {name.surname}@unibz.it

Muhammad Ovais Ahmad, Kari Liukkunen
University of Oulu
Oulu, Finland
e-mail: {name.surname}@oulu.fi

Ilaria Lunesu, Martina Matta
University of Cagliari
Cagliari, Italy
e-mail: {name.surname}@diee.unica.it

Fabian Fagerholm, Jürgen Münch
Department of Computer Science, University of Helsinki
Helsinki, Finland
e-mail: {name.surname}@cs.helsinki.fi

Sami Pietinen, Markku Tukiainen
University of Eastern Finland, School of Computing
Joensuu, Finland
e-mail: {name.surname}@uef.fi

Carlos Fernández-Sánchez, Juan Garbajosa
Technical University of Madrid (CITSEM & ETSISI)
Madrid, Spain
e-mail: carlos.fernandez@upm.es

Kari Systä
Tampere University of Technology
Tampere, Finland
e-mail: kari.systa@tut.fi

Abstract— **Entrepreneurs and Small and Medium Enterprises usually have issues on developing new prototypes, new ideas or testing new techniques. In order to help them, in the last years, academic Software Factories, a new concept of collaboration among universities and companies has been developed. Software Factories provide a unique environment for students and companies. Students benefit from the possibility of working in a real work environment learning how to apply the state of the art of the existing techniques and showing their skills to entrepreneurs. Companies benefits for the risk-free environment where they can develop new ideas, in a protected environment. Universities, finally benefit from this setup as a perfect environment for empirical studies in industrial-like environment. In this paper, we present the network of academic Software Factories in Europe, showing how Companies already had benefit from existing SFs and reporting success stories. The results of this paper can increase the network of the factories and help other universities and companies to set-up similar environment to boost the local economy.**

Keywords—Software Factory; Experience Report.

I. INTRODUCTION

Universities are perfect environments to exploit technological research for innovation. The biggest challenge to solve in universities is that they are nowadays rarely used by companies, and at the same time, universities are poorly oriented to give economic value while start-ups and Small and Medium Enterprises (SMEs) face new and tough challenges to survive in the market. Indeed, also big industries sometimes have difficulty being continuously innovative. In fact, ideas come out slowly and require a lot of effort to be implemented.

Fresh ideas, coming from the new digital native generation of developers, should encourage seniors to fresh thinking. From this aspect, the combination of university research, teaching and industry production would increase the value of skills of everyone and the development of innovation. Innovations lab of similar initiatives such as academic Software Factories (SF) [1] could contribute to fill this gap.

In SF, students and entrepreneurs collaborate together to develop a new idea or to apply existing techniques that couldn't be achieved by the entrepreneur itself without accessing to external resources. SF are university laboratories that emulate a real working environment, in which a given number of students, in the same location, work as a real team implementing a project for 7-11 weeks in a controlled environment with real customers and real deadlines. Entrepreneurs benefit from the new innovative ideas and the effort coming from students. Students have, in turn, a unique experience of working in an industry-like work environment getting in touch with the real business and a given number of credits. Moreover, students have the chance to present their skills to the entrepreneurs that can finally hire students partially trained on their technologies. The SF initiative is a fully bottom up initiative that cooperates on a voluntary basis without any funding framework, except for their enthusiasm and the common interest in getting excellent educational, and research results.

The goal of this work is to present the ecosystem of European SFs with a set of success stories. Moreover, we aim at analyzing similarities and differences among SF in different countries (namely Finland, Italy and Spain), highlighting pros and cons for the different stakeholders. The results of the paper show how the use of SFs, as safe environment for developing new prototypes and products for start-ups or entrepreneurs, could represent a good practice and an important starting point for creating a connection between academic and working worlds.

The paper is structured as follows. After a first introductory section, we describe the SF concept in Section II. In Section III, we describe our international SFs, highlighting similarities and differences. We report success stories in Section IV and finally we draw conclusions and future works in Section V.

TABLE I. SF BENEFITS

Academic Institutions	Companies	Students
• Perfect environment for empirical studies • Provide better training to their students. • Collaboration with industry • Environment for the development of research prototypes	• Environment to develop innovative ideas. • Environment to test new development tools or methodologies. • Opportunities for hiring new staff trained in the technologies that they use.	• First early contact with real-world projects • Develop transversal capabilities such as self-organized, responsibility, communication, etc. • Put into practice the theoretical concepts learned in the courses • Learn new techniques and technologies.

II. BACKGROUND

SF proposes itself as an infrastructure that supports research and education in software engineering and also entrepreneurship. In the Finnish editions, many collaborations with important organizations guided forward good results for customers and developers. SF is a safe and monitored environment that reproduces in a faithful manner the working team dynamics that develop a prototype or a software product for a customer, (SMEs) or a start-up. Since its first edition SF brings together three essential goals: Learn, Share, Grow. SF [1] represents also a shared educational platform for universities to hold courses where students are involved in a real-world project developing software in the same location or in different sites. SF relies on self-organization as its primary way of organizing the work [2].

It represents a unique platform in which a team of students develops software. SF projects are conducted in a manner that simulate as closely as possible, the reality of software development in the product development organization. We can then observe how SF could represent, despite its constraints and limitations, the operational core from which startups, entrepreneurs or SMEs could set up their own ideas allowing, at the same time, smart and brilliant students to make a unique practical experience learning by doing new methodologies and practices but also approaching the working world through the main door showing what they can do.

In addition, SF offers a way to learn new practices and technologies not only by reading from books but also by building a product. The results are achieved as a result of collaborative work of all team members, to improve their knowledge and skills getting in contact with people having different background and experience. As the students need to independently gain new knowledge and meet new people to create the product they get in touch with working reality and undertake new important collaborations. At the same time, SF are independent and open for collaboration with all kind of companies for entrepreneur or startups, the SF could represent a low cost environment in which they can set up new ideas and new projects in which create not only a prototype but meet partners and developers to be integrated in their own team having the advantage of a training period. The SF advantages exist also for researchers or academic members that would like to have the possibility to assist to the meeting of two worlds: work and study but in a monitored lab environment. This fosters the measures and observations to make research from a software perspective making measure about effort or software metrics and also from the educational perspective observing the student interactions, their learning behavior and their attitude in the creation of a new product. Table I summarizes the benefits of the SF environment.

III. THE SOFTWARE FACTORIES NETWORK

In this section, we describe the SF network in Europe presenting the different set-up and operational model.

A. The Helsinki SF (Finland)

SF at the University of Helsinki [1][7] has been organized since 2010. The factory deeply integrates the customer company into the development process. The customer provides a product owner who interacts directly with the student team during the project. The customer can range from local entrepreneurs to large enterprises and even to Open Source projects. For example, in spring 2015, the factory is participating in Facebook's Open Academy program, collaborating on two Open Source projects with universities worldwide. As a rule, five projects are arranged per year. The factory supports the projects with research-based insights for project management, methods, and pedagogy, and through full-time coaching of the teams.

B. The Bolzano-Bozen SF (Italy)

The factory [6] is organized by the Free University of Bolzano-Bozen. It is actively running once a year for 4 years, developing more than 10 projects. The participants are students from the first year of the Master program in Computer Science, third year of the Bachelor in Design and Education faculties, as well as local entrepreneurs. Project ideas come mainly from local entrepreneurs who are not affiliated to the university. The course runs during the summer semester for 11 weeks with a required effort of 200 hours per student. Students vote the projects to be developed based on their interests and skills. The most voted projects are developed during the SF. Students are then split in groups of 5-6 people and every group is assigned to a project. The entrepreneur who proposed the idea is required to be available at the SF at least once a week to support the students.

C. The Cagliari SF (Italy)

The factory has being running once a year for 3 years from 2012, developing a total of 4 projects. Participants are students from the Master program in Electronic Engineering, Telecommunication Engineering, Computer Science, PhD Course and local entrepreneurs. Projects come local entrepreneurs or ideas born for implementing applications to satisfy the needs of the research group. The course runs during the summer semester for 7-11 weeks with a required effort of 120-200 hours per student, with 4-8 people assigned to each project. The entrepreneur who proposed the ideas is required to be in class to support the students at least twice a week. The development is driven by an expert PhD student that plays the role of coach/coordinator. In order to replicate a real company development environment, an open space is assigned to the

team and the team members have to come twice or three times at week during the period.

D. The Joensuu SF

University of Eastern Finland's School of Computing established SF Joensuu in 2010 and is running 3-5 rounds per year. Teams consist of mostly master level computer science students with minimum target of 4 people. They are encouraged to participate two rounds, first round as software developer and later, second as team leader. Product ideas come from entrepreneurs and research groups with having target to produce new business opportunities or improve the world in general. SF team is supported by mentoring given by SF lead and students from previous rounds. Frequent interaction with customer is required in order to achieve release cycle of 1-2 weeks, preferably with face-to-face meetings at least at same interval and with other medium more frequently. Customers have been either starting entrepreneurs with just a good idea in their hands or already established companies from start -ups such as Epooq to big companies like CGI.

E. The Oulu SF (Finland)

The Oulu SF is established in 2012 to provide a realistic environment, which improves the students' learning experience by providing them with insights into the conduct of real-life software projects with close customer involvement, intensive teamwork, and the use of modern software development tools and processes [5][6]. As a platform, it serves multiple purposes. It is a test bed for software engineering ideas and a source for original basic scientific software development research. Oulu SF runs twice a year and it has completed more than 8 projects since 2012. The participating students are from first and second year of master's degree in information processing science. The project tasks come from the local software companies and or research projects. Each project involves a minimum of four members. The students are encouraged to tackle management and resource planning issues pertaining to large teams. Each project team is assigned a project supervisor who provides the team with technical and non-technical guidance. The supervisor is also responsible for monitoring and assessing the team throughout the course of the project.

F. The Madrid SF (Spain)

Madrid factory has been operating since 2011. The factory was a joint set-up between the Technical University of Madrid (UPM) and Indra Software Labs, a subsidiary of Indra, a Spanish global company. Actually, two software factories were set up, one in UPM and one in Indra Software Labs, to run joint projects. As an educational setting, students that participate are from a Degree in Software Engineering and Masters Program on Computational Science and Technology. Most of the projects are closely related to tasks of European or National Research projects, very often collaborative projects between industry and UPM. Different kind of projects have been performed over these years, and more than 10 projects: five of them were distributed projects, in which up to three nodes from different countries were involved, e.g., Helsinki (Finland) and Bolzano (Italy). The Factory recruits students usually in November and February. The recruitment process includes an interview to applicants with a given number of questions. Even when some these questions are technical, other skills are also searched, also considering the kind of project that the student would like to be involved, and the positions available for each project. Students usually work in teams, from 3 to 6 students. Students perform slots of 140 hours, during generally 8 weeks, and up to a maximum of 3 of these slots.

IV. Similar Activities in Tampere (Finland)

A project course with many similarities to SF has been run at Tampere University of Technology from year 1991. Already during the first years, we received project ideas from companies, and this collaboration has been a key component of the of project course. Currently, the project ideas come from the companies and companies give constant feedback about progress of the project and produced software. These companies essentially play a role of customer for the student team. Since the course has a long history in Tampere, many managers in the surrounding companies have participated in the course in the past, they now have a high motivation to collaborate as a customer. Key role of the companies is seen valuable for both students since it gives both parties an opportunity to network.

In addition to independent project work, the course also includes some lectures that help students in management of the project. There are also lectures about IRR, legal and business aspects. The volume of the project course has been varying over the years. During academic year 2014-2015 there were 10 project teams with 5-8 members each. The number of hours spent per student is in the range of 130 – 260 hours. The course is run yearly starting September and ending in February to an end seminar and celebration.

Since 2008, Demola [8], has also been another option in Tampere. Demola focuses on innovation projects, where students are asked to further develop ideas given by surrounding companies and public institutes. Demola is hosted by Hermia (a business development company) and three universities, Tampere University of Technology, University of Tampere and Tampere University of Applied Sciences, participate in Demola. The project teams are cross-disciplinary and consist, e.g., of engineering, business, and design students from participating universities. Since Demola projects concentrate in innovation and further development of the idea, the process includes value creation workshops and pitching events. Furthermore, Demola shares promises with Protomo [9], which is a development community for new businesses and start-ups

Regular Demola projects run twice a year: once in the fall and once during the spring. The volume of 20-25 projects per season typically run by groups of 4-5 students. The main difference between Software Engineering Project Course and Demola is that the former concentrates in professional development projects while the latter concentrates in idea development and innovation.

V. Software Factory Success Stories

Software factories actively supported local entrepreneurs. Here, we report on some of the most important success stories of our factories.

A. Innovative Video Calling Service

Between September 2012 and December 2013 the SF Helsinki conducted three projects together with Tellybean Ltd., a small Finnish startup [3]. The vision of this startup was to deliver a life-like video calling experience for specific customer segments such as elderly persons. The overall goal of the collaboration was to conduct build-measure-learn loops to validate critical assumptions underlying the business model and the technical solutions in order to rapidly learn if the chosen strategy needs adjustments or can be persevered. The first project focused on the development of appropriate analytics for measuring the performance of the video service so that business-critical information can be gathered and analyzed. In addition, technical feasibility aspects were analyzed. The second project mainly focused on validating technical assumptions. In particular, the company wanted to understand the scalability and robustness of the proposed system architecture, technical weaknesses of the system, and the company wanted to identify alternative options for the system architecture. The project resulted in a significantly better understanding of the limitations and future development options. The third project helped the company to better deploy functionality in a continuous way.

Overall, the prototypes that were created and used in the projects served as so-called minimum viable products to quickly validate business-critical assumptions and helped the startup to accelerate learning about their ideas. In the meantime, the startup got significant funding. Now, Tellybean partners with major service providers. The SF Helsinki benefited well from these projects by learning how to organize industry-academia collaborations in order to test business-critical assumptions.

B. Memoree

Memoree was a SF project at Bolzano in spring 2013. It was based on a business idea from a local entrepreneur who needed to develop a prototype to prove his idea. The initially intended software solution would pack personal photos, videos and audios into a memory package and shared it among friends. The project lasted 11 weeks. In total, 14 students were involved in the Memoree project. The majority came from the Computer Science faculty. Two designers were involved at the later stage of the project. The entrepreneur played the customer role for the project and made himself available all through the SF session. The Memoree SF project was very useful for the local entrepreneur to understand what are the crucial features of Memoree, and what should be skipped. SF also helped him to decide what could be the core component of the application. The developed prototype was very different than the initial idea that he had. The students were not just implementing the prototype, they were contributing to the understanding of the need the startup intended to meet, and the clarification of the vision that drives the startup. After the SF session, the Memoree idea became more concrete. It is positioned as a mobile application that is developed for automatic creation of videos based on different contents (photos, songs, etc.). The app provides content privacy, and creates videos automatically by taking songs and photos as an input. This application is composed of two modules: content management (photos) and video creation. The intended customers are iOS users. The entrepreneur team was expanded from a single person to five founders (two economics, one finance, one graphic designers and one computer scientist). The actual development started in May 2014.

C. Medygo

Different from the Memoree case, when a founder of Medygo approached the Bolzano SF in Spring 2014, the business idea has already been validated initially with potential customers, and a prototype was developed already. It is a mobile application that is developed with motive “health on go”. It is mainly developed for people to solve their health problems during traveling and staying abroad. The main purpose of developing this application was to prevent travelers from the hustle when they travel and become sick during their journey. This mobile app converts medicine, what they take in their own country, to what they should take in another country. This app is initially developed for android users. There were four founders (two businessmen, two pharmacists). It’s been one year since they have been working on this idea before they contacted the Bolzano SF. The actual development started in November 2013 by adding another team member as a developer. The initial version was launched in January 2014.

One of the benefits of working in SF was that, recalled by one founder, was the iterative approach the SF adopted. There were always some deadlines, and the team had to finish on time. During the SF session, the Medygo team set milestones e.g., two-week idea validation, two-week date collection, four-week development and so on. At the end of the 10th week, their prototype was ready. The team was also facilitated by the SF tutors to handle pressure, and to meet deadlines. In addition, the SF students worked on the project became potential hire for the startup company due to the intimate knowledge they obtained through working on the project at the SF.

D. Matchall2

The Matchall2 project was proposed by a local entrepreneur that played the customer role during the development period, with the aim to build a plugin for categorizing personal multimedia content gathered from famous social networks such as Facebook, Youtube and Flickr. Matchall2 created a personal communication engine based on innovative principles and functionalities, with a web implementation and diffusion strategy. The final prototype, thanks to an idea of some developers, was represented by a bookmarklet that allowed one to easily classify and categorize personal content, such as pictures and videos, in a customized manner using tags. The focus of the project was to implement the same application for different social networks. This SF started in early March of 2013 and held 11 weeks involving 8 specialized students with rich and different skills and backgrounds.

In order to take advantage of developer's skill diversity, the development was organized considering pairs in which an expert developer supported a less experienced student.

In this edition, many technologies and new abilities are used and learned to obtain the maximum result. The meetings with the entrepreneur allowed to stay in the right edges of time and specifics. The particularity of this case is that people also when they don't know each other thanks to the fact they have the same aim, strive to implement a success product, behaving

as a family helping each others to solve problems or achieve the same objectives.

E. SERTS (Software Engineering Research Tool Suite)

During the SF of 2013 edition, the project Software Engineering Research Tool Suite (SERTS) has been developed by a team of 6 students. The aim was the implementation of a semi-automatic tool able to semplify the analysis of data collected in software repositories such as Bugzilla, CVS, SVN, Git, and Jira. The development period lasted eight weeks, from September 2013 to November 2013 by a team composed by six developers: a PostDocstudent, four PhD students and one undergraduated. In this specific project, a medium knowledge of software development was required. Each component of the team had different tasks, chosen according to their skills. One of the PhD students with a strong knowledge of the technologies involved into the project, played the role of team coordinator/coach. The used development process was Scrum with iterations of two weeks.

The figure of the customer was very significant. Every two weeks he monitored the work of the students observing the progress of the project through spikes. Due to its constant presence, it was possible to build a prototype inherent to its requirements.

F. FREI MARKT SÜDTIROL

In 2014, Oulu SF and Bolzano SF start a collaborative time banking project "FREI MARKT SÜDTIROL". An Italian entrepreneur was sending requests for his project to Oulu SF team. The project idea was to provide a common single plate form for existing time banking systems in South Tyrol and other near cities.

Project aim was to provide a fresh new time bank-community system which cover various parts of society and particularly for those people who are strongly hit by the ongoing socio-economic crisis including young unemployed, working-poor and immigrants. A system was developed which allows users to create their personal profiles, look for jobs & products, post jobs & products, apply for jobs & product and give feedbacks. In addition, a SMS platform will facilitate the new member registration process, modification and verification of time-checks (BiX) when the people are not familiar with the Internet. Project team consists of eight students; in which four students working from Oulu SF and four from Bolzano SF. Both teams were having mentors to help agile and lean concepts in the project. The use of Kanban method and JIRA was mandatory for Oulu SF students while the Bolzano SF students were not following any specific methodology or practices. Both teams use and get experience with Rise Editor, Myeclipse, Apache Tomcat, PostgreSql, Dreamweaver, and GitHub in the project. In first two weeks, students attend mandatory lectures and exercises in SF. During weeks 3 to 5 literature was studied related to the project idea, working methodology and preliminary project plan were drafted. Then design and actual development related tasks were carried out within weeks 6 to 12. After every two week the teams deliver batch of minimum viable product to customer. The project demo was given in Bolzano University which was appreciated and covered by local press.

G. Google Glass for Traffic Warden

One of the latest Demola examples is a project where five students got the idea from a local SW company Vincit but also collaborated with City of Tampere. The project integrated automatic recognition register plates to Google Glasses. In this project, the student group developed the first commercial smart glass application in Finland. With this application the traffic warden is able to see right away if the parking ticket has been paid or not. The city of Tampere is piloting this system in spring time 2015 [13].

Development of this system may not have been possible with traditional processes where software companies and public authority as customer should recognize the idea first and then have detailed enough specification. In this case, the student group approached the idea as a start-up by trying and doing. The system project received also a fair amount of publicity in Finland.

H. Optimeter

The Optimeter project was developed in 2012 by the Madrid SF (Technical University of Madrid and Indra Software Labs) and the Helsinki SF (University of Helsinki). The Optimeter project (in practice there were two projects inside the SF, Optimeter I and the subsequent Optimeter II) had as goal to implement some use cases about data acquisition in intelligent power networks, usually known as power smart grids. The objective was to build a benchmark to validate massive raw data coming from sensors and smart meters. The benchmark was created using Apache Hadoop and Oracle NoSQL Database to provide distributed processing and storage capabilities to the system. Optimeter I and II were traversal to two European projects under the ITEA2 Programe: IMPONET (Intelligent Monitoring of POwer NET [10], 127 man years) and NEMO&CODED (Networked Monitoring & Control Diagnostic for Electrical Distribution [11], 112 man years, and a third Spanish project called ENERGOS (Technologies for automated and intelligent management of power distribution networks of the future [12], with a budget of 24.3 million euros). The project was developed using agile practices, and more concretely following the Scrum methodology. Optimeter was an excellent framework to set up a collaboration activity between three Software Factories (UPM and ISL in Madrid, and UH in Helsinki).

One lesson learned is that the training that the students can get in such environment is very useful but straining. Students were under the same pressure that the industrial development team during the weeks the project took place. But at the end, the background, skills and experience were very much welcome by the students.

From the point of view of the industry, they could develop the software that they needed, experimenting the usage of agile methodologies in a distributed development environment. Also they use the project to test some development tools that they were not using until that moment.

I. HavuSport

In 2013, School of Computing in Joensuu was contacted by two hockey coaches with an idea that the junior coaching should be supported by an electronic system, usable with different end user devices. Mobile device is the device of the

day that younger people easily relate to. The system should support all major activities of a hockey team from messaging and timetabling to performance statistics while getting rid of excel-sheets, paper and pen. They could not find a proper existing system, so they decided to have it build. SF Joensuu built a web-based system with mobile applications coming aside during two project rounds. Team size was 8 people, but the project needed to be scoped very well because of the high amount of required features and the fact that there was a lot to learn in a short period of time.

Team felt a real business pressure to deliver, a feature that is build inherently to SF, succeeding to achieve the target in time. They felt proud that their hard work paid off. Havusoft Company Ltd. was formed around the product and now Havusoft is planning to extend the software system for other sports activities too and there is great interest in the market to use the system. The major role of SF concept in this process was to enable starting entrepreneurs to push their idea forward and show to the world that they are serious with their endeavor.

VI. Conclusion

In this paper, we present the academic Software Factories (SF) in Europe, describing how they can help the local economy by means of the collaboration among academia, entrepreneurs and SMEs.

Our goal is to report on our SFs and similar initiatives, presenting success stories.

SF provide an unique environment where entrepreneurs can explore new ideas, develop new prototypes or apply new techniques and students can study and work in a setting that replicate, as much as possible, a real work environment. Moreover, students have the opportunity to show their skills to entrepreneurs and entrepreneurs can find new developers easily, based on a direct knowledge of the students itself.

The network of SF in Europe, shared among Finland, Italy and Spain is composed by several University that serves hundreds entrepreneurs. SF collaborates with some shared projects, working for the same project collaboratively, such as the project described in the success story “Frei Markt Sudtirol”. We reported several success stories, such as the Google Glass for Traffic Warden, Memoree, Optimeter and others, showing how the different stakeholders benefit of the SF environment, from an entrepreneurial, didactical, and research points of view.

A new Software Factory has recently been established at Montana State University in Bozeman, MT. A first project with a company from the financial sector has started and relationships with entrepreneur communities such as Blackstone Launchpad have been established.

In the future we plan to further expand the community extending the number of partners’ universities and increasing the number of shared projects and involving industries in the project selection and execution.

Acknowledgments

The authors would like to thank the companies and their employees for participating to this research. This research has been carried out in Digile Need for Speed and Digital Services programs, and it has been partially funded by Tekes (the Finnish Funding Agency for Technology and Innovation), the Italian Regione Autonoma della Sardegna (RAS), Regional Law No. 7-2007, project CRP-17938 LEAN 2.0, the Spanish projects iSSF (i-Smart-Software-Factory) IPT-430000-2010-38, INNOSEP TIN2009-13849, IMPONET ITEA 2 09030 TSI-020400-2010-103, NEMO-CODED ITEA2 08022 NEMO CODED IDI-20110864, and ENERGOS CEN-20091048.

We also thank Ville Korpiluoto from Demola (Tampere) and Xiaofeng Wang (Free University of Bolzano) for reviewing and supporting this paper.

References

[1] P. Abrahamsso, P. Kettunen and F. Fagerholm, ”The set-up of a software engineering research infrastructure of the 2010s.” In Proceedings of the 11th International Conference on Product Focused Software ACM. pp. 112-114, 2014.

[2] X. Wang, I. Lunesu, J. Rikkila, M. Matta and P. Abrahamsson, ”Self-organized Learning in Software Factory: Experiences and Lessons Learned”. In Agile Processes in Software Engineering and Extreme Programming. pp. 126-142, 2014.

[3] F. Fagerholm, A. Sanchez Guinea, H. Mäenpää and J. Münch, “Building Blocks for Continuous Experimentation”. In Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering (RCoSE 2014), Hyderabad, India,. pp 26-35, June 2014.

[4] M.O. Ahmad, K. Liukkunen and J. Markkula, J., “Student perceptions and attitudes towards the software factory as a learning environment”. IEEE Conference on Global Engineering Education. Istanbul, Turkey. pp 422 – 428, 2014.

[5] M.O. Ahmad, J. Markkula and M. Oivo, “Kanban for software engineering teaching in Software Factory learning environment”. World Transactions on Engineering and Technology Education (WIETE), Vol.12, No.3, pp 338-343, 2014.

[6] Bolzano-Bozen Software Factory, http://www.newsoftwarefactory.org (Accessed: June 2015).

[7] Helsinki Software Factory, http://www.softwarefactory.cc. (Accessed: June 2015).

[8] Demola. www.demola.fi. (Accessed: June 2015).

[9] www.protomo.fi. (Accessed: June 2015).

[10] IMPONET, https://itea3.org/project/imponet.html (Accessed: June 2015).

[11] NEMO&CODED, https://itea3.org/project/nemo-coded.html (Accessed: June 2015).

[12] ENERGOS, http://innovationenergy.org/energos/ (Accessed: June 2015).

[13] Google Glass for Traffic Warden. http://googleglassfortrafficwarden.blogspot.fi (Accessed: March 2015).