

Development of Agile Practices in Romanian Software Community

Eduard Nicolae BUDACU
 Bucharest University of Economic Studies
 eduard.budacu@csie.ase.ro

Agile Software Development (ASD) promotes flexibility to change and emphasis the importance of individuals and interactions in producing software. The study presents the development of agile practices in Romanian software community. A literature review is conducted and the main agile methods are described. The characteristics of Romanian ICT sector is presented in relation with agile methodology. Practices are identified by a survey and an analysis on the groups of interests formed on Meetup website is performed. Future directions and development of agile practices is evaluated.

Keywords: Agile Practices; Software Development; Scrum; Kanban; Extreme Programming

1 Introduction

Agile Software Development (ASD) methodology is defined by four values and twelve principles that sum up the essence of a variety of methods and practices. A certain attitude towards building software is promoted that puts individuals and interactions before processes and tools. Agile is a new way to see the software development process by taking in account human characteristics. In order to produce working software people need to collaborate towards common goals and find solutions that would maximize the value created. Spending too much time to develop comprehensive documentation is considered waste. Collaboration between the development team and business is important and favoured to contract negotiation. This means optimizing any process in order to create a system that is based on transparency, inspection and adaptation that leads to reduced bureaucracy. Flexibility and responsiveness to change is preferred over following a plan.

The ICT industry in Romania has known a significant growth in the past years and became an attractive option for providing outsourcing services. This paper aims to assess the development of agile practices and evaluate how they are evolving to sustain the growth of the Romanian ICT sector. Reasons for adopting agile practices are [1]:

- Accelerate product delivery;
- Enhance ability to manage changing priorities;
- Increase productivity;
- Enhance software quality;
- Enhance delivery predictability;
- Improve business/IT alignment;
- Improve project visibility;
- Reduce project risks;
- Improve team morale;
- Improve engineering discipline;
- Reduce project cost;
- Increase software maintainability;
- Better manage distributed teams.

The main agile methods that gained popularity are presented in Table 1.

Table 1. Content details

Method	Authors	Year	Ref.
Scrum	Ken Schwaber, Jeff Sutherland.	1995	[2]
Extreme Programming (XP)	Kent Beck	1999	[3]
Kanban (development)	David J. Anderson	2010	[4]
SAFe: Scaled Agile Framework	Dean Leffingwell	2011	[5]
LeSS: Large Scale Scrum	Craig Larman, Bas Vodde	2013	[6]

Scrum is the most popular of the agile methods. It was initially presented at the OOPSLA conference in 1995. In the original guide published by the authors Scrum is presented as a framework for developing and sustaining complex products. The definition consists of three roles, four events, three artefacts, and the rules that bind them together. Scrum framework values are:

- Courage;
- Commitment;
- Focus;
- Openness;
- Respect.

The Product Owner is responsible for maximizing the value of the product and the work of the team. The Development Team consists of professionals who do the work of delivering a potentially releasable Increment of “Done” product at the end of each Sprint. The Scrum Master is responsible for ensuring Scrum is understood and enacted. Scrum Masters do this by ensuring that the Scrum Team adheres to Scrum theory, practices, and rules.

The Sprint is a time-box of one month or less during which the team aims to deliver an improved version of the product. The Sprint starts with a collaborative meeting called Sprint Planning where the Product Owner presents priorities from a Product Backlog and the Development Team helped by the Scrum Master figures out what can be accomplished. The Daily Scrum is a 15-minute time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours. A Sprint Review is held at the end of the Sprint to inspect the new version of the product and adapt the Product Backlog if needed. The Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.

Extreme programming (XP) is a software development methodology which is intended to improve software quality and responsiveness to changing customer requirements. It uses the same dynamics of Scrum by defining short iterations (similar with sprints) and extends the framework with rules that aim to enhance technical excellence. In order to assure high

quality code must always be written to an agreed standard, unit tests must be created and pair programming is recommended for all production code to increase collective ownership. Kanban uses principles from Toyota Production System (TPS). The method aims to create a continuous stream of work by identifying blockers in the process. The core practices as identified in [4] are workflow visualization using boards, limiting work in progress, manage the flow, make process policies explicit and continuously improve the system. Using this practices leads to an increase in the transparency within the team and assures that impediments are removed as soon as they emerge. Kanban doesn't prescribe any specific roles. Rather than making massive change in the organizational structure an evolutionary approach to process improvement is preferred that would reduce the risk of resistance to change.

SAFe: Scaled Agile Framework and LeSS: Large Scale Scrum is designed to ensure agility in an enterprise context with multiple teams that work on the same product. Adaptation on Scrum framework is proposed to assure the coordination of projects and utilization of shared resource when there is more than one team involved.

2. Research method

The main goal of the research was to develop a solid understanding on software development methodologies from both a theoretical and practical point of view in order to design a learning program that will increase efficiency of the software teams within a company. A multi-method approach was considered for this study in order to address the subject from different angles.

A literature review was conducted in order to understand the theoretical aspects of Agile Software Development:

- Books from authors that are considered thought leaders in the field were selected;
- Scientific papers that consist in systematic literature reviews on agile topics were consulted [7], [8], [9];
- Study cases, surveys and market studies provided by main vendors of agile tools

were analyzed.

In order to exchange experience with other practitioners involvement in the groups of interest formed on Meetup.com was considered. Working with teams to improve the process and increase performance provided insights on how to translate the theory into practice.

A survey was designed to gather quantitative data. The survey offered an overview on the agile methods used in Romanian software teams. The content of the form was designed using reference questionnaires like: [1], [10], [11]. Furthermore, four interviews were held with persons involved in agile programs, to verify that the agile theory and the previous mentioned forms used as guidelines capture the reality of agile in software teams.

The tool used was Google Forms and three weeks were allocated for collecting data. The target audiences were persons who worked in software teams and certified agile practitioners. The channels used for distribution were:

- www.meetup.com platform in the communities that had an agile interest stated;
- direct email to companies from www.softwareindex.ro (approximately 50 companies);

- on social media channels to several group of interest (www.linkedin.com and www.facebook.com);

Preliminary results were presented during the IE 2015 Conference [12].

An important method to expanding knowledge regarding agile practices was participating in events organized by groups of interest that share the same concerns and meet regularly to exchange experience. These groups use the social network Meetup to organize events where case studies, best practices and personal experiences are presented. Participation to events is free of charge and usually takes place in co-working hubs or at company's headquarters.

Scrum courses were taken with trainers certified by Scrum Alliance. With the theoretical foundation and having previous experience in working in software development companies the role on Agile Coach was undertaken to gain more experience in agile practices. The role was initially defined in [13]. Case studies of companies adopting agile are presented in Table 2.

Table 2. Summary of agile case studies

Company	Duration	Program goals	Role
ALPHA	July 2013 – January 2015	Improving the efficiency of the development process	Department Lead
BETHA	Mai 2015 – August 2015	Knowledge transfer on agile methods and practices	Agile Coach
GAMMA	September 2015 – December 2015	Improving teamwork using Scrum	Scrum Master
DELTA	March 2016 – May 2017	Increase agility for the IT department in order to support company growth	Agile Coach

ALPHA is a consultancy company providing end-to-end services like software development, design, quality assurance, online marketing, technology consulting and ecommerce solutions. As client projects become more complex the need of improving the efficiency of the development process became evident.

Company BETHA is a small agency delivering web-based systems and maintaining in-house developed educational products. The main challenge included working with state of the art technology in order to deliver a functional prototype for a complex form builder. Having a small team people were allocated on multiple projects so having a flexible way to

organize the work was required. Company GAMMA develops and maintains a platform that connects online retailers with affiliates that provide marketing services. Launching a new version of the platform involved increasing the development team so the program focused on improving teamwork by using Scrum.

DELTA is an ecommerce startup based in New York that has the IT development department in Bucharest. Due to rapid growth of the company changes in the organization of the IT department were considered to increase agility and improve coordination between software teams and business teams.

All companies considered Scrum as framework for improving teamwork and gain more agility in the development process. Management was willing to allocate resources to improve the software development methodology in order to achieve better results.

3. Results

Each method approached generated valuable data and the results were processed within visual graphs, tables or a series of observation. The survey has reach around 1000 people, based on the estimated size of the target group. A total of 358 people opened the link and 119 completed the form. Table 3 presents the results of the survey.

Table 3. Agile Practices in Romania

Category	Practice
Agile Method	Adaptive software development (11.9%) Agile Modeling (15.3%) Agile Unified Process (3.4%) Crystal Clear (2.5%) Disciplined Agile Delivery (9.3%) Dynamic Systems Development Method (2.5%) Extreme Programming (28.8%) Feature Driven Development (19.5%) Lean software development (14.4%) Scaled Agile Framework (5.1%) Scrum (90.7%) Scrum/XP Hybrid (22%) Scrumban (18.6%)
Project Management	Cross-functional team (59%) Definition of Done (66.7%) Acceptance Criteria (57.3%) Burndown Chart (54.7%) Burnup Chart (14.5%) Kanban Board (43.6%) Task Board (42.7%) Planning Poker (51.3%) Product Backlog (83.8%) Product Roadmap (44.4%) Sprint Backlog (79.5%) Story Mapping (30.8%) Story-driven modeling (13.7%) Time-boxing (41%) Use case (34.2%) User Story (76.1%) Velocity Tracking (50.4%) Business Canvas (0.9%)

Team Meetings	Sprint Planning (87.2%) Daily Standup (85.5%) Sprint Review (72.6%) Sprint Retrospective (78.6%) Backlog Refinement (49.6%) One-on-one meeting (42.7%)
Engineering Practices	Acceptance test-driven development (24.6%) Agile Modeling (20.4%) Agile Testing (45.6%) Agile Software Architecture (27.2%) Behavior-driven development (17.5%) Continuous integration (64%) Continuous deployment (35.1%) Domain-driven design (13.2%) Iterative and incremental development (28.1%) Pair-programming (45.6%) Refactoring (69.3%) Code review analysis by peers (73.7%) Static code review analysis with tools (32.5%) Test-driven development (49.1%) Version control system (67.5%)
Knowledge Sharing	Company Blog (23.9%) Communities of Practice (27.4%) Informal discussions (69.2%) Internal presentations (74.4%) Mailing lists (41.9%) SharePoint sites (22.2%) Technical meetings (62.4%) Training programs (51.3%) Workshops (40.2%) Wiki (53.8%)
<i>Learning Methods</i>	Books (70.4%) Communities of practice (33.9%) Contracted consultant (10.4%) eLearning (33.9%) Full-time agile coach (9.6%) Internal agile support group (29.6%) Training program/workshop (47.8%) Self-study (81.7%)

The Meetup community was analyzed by extracting the data from the platform via their API. Figure 1 depicts the five main ICT hubs in Romania. The number represents how

many groups with concerns in software development and technology are active in each location.



Fig. 1. Number of Meetup groups sharing interest in agile practices

The majority of groups were founded between 2014 and 2016. Some of the oldest groups include The Bucharest Agile Software Meetup Group (2008), Tabara de Testare Bucuresti (2012) and Lean Startup Bucharest (2012). For example *AgileWorks* is a community of practice with the purpose of spreading Agile and Software Craftsmanship best practices in Romania. It was founded in 2008 by Maria Diaconu. Members in the community are practitioners from software companies like developers, testers, project managers, agile coaches, scrum masters, trainers. The group organizes events that aim to bring together individuals with the purpose of socializing and sharing experience. The community has multi-disciplinary approach with subjects of interest varying from coding, technical best practices, software craftsmanship, agile methods, agile coaching and soft skills. Meetings are organized with the involvement of the community members. The most popular event called *Agile Talks* is hosted every month by local tech hubs or companies and brings two

speakers that share their experience and learnings. Group discussions are facilitated after the presentation using the Open Space technique.

Company ALPHA has switch from a resource allocation model based on client needs to a model that relies on dedicated teams that work in an agile way. Each project team had a project manager that worked closely with the client to figure out requirements, several software developers ranked by seniority and software testers. Resources like graphic designers and systems administrators were shared between teams. The delivery model still relied heavily on process to assure that the same standards of quality were respected across teams. Scrum was used to organize the work. Initially sprints were planned and managed by using a software developed in house and an open source issue tracking system. After several trainings with an external consultant Scrum events were introduced that led to more face-to-face communication and improved coordination between team members. As an effect project deadlines were met more often

and the client got involved in the process to provide valuable feedback for the teams. A new project management tool was implemented to reflect the changes in the workflow and a continuous integration system that performed static code analysis was developed. Having a small team working on multiple projects a full agile method could not be implemented for company BETHA. Certain practices were adapted to improve how work was organized. Initially a product roadmap was created to identify specific phases of the project. Two weeks sprints were proposed. Because work was done on multiple projects planning was hard and constant changes in the scope lead to unfinished work at the end of the sprints. In order to increase transparency a Kanban board was displayed in the room. While some practices were considered useful in helping the team getting more organized a formalized method could not be implemented due to nature of the projects. Work was done on an individual level based on direct relationship with the customers. Agile methods are suitable when the group shares common goals.

The software team in company GAMMA was already using some of the Scrum practices when the program begun. An online board was used to keep track of tasks, sprints were planned and features were estimated using story points. After several discussions to understand the company we started collaboration in the role of Scrum Master to help facilitate Scrum events, improve agile process and implement JIRA as a project management tool. Several milestones with the new product were reached but in order to increase the speed of delivery new developers were hired and certain roles were covered. A product manager was hired to deal with the product backlog, help the developers understand requirements and plan the launch of the new version. A software tester joined the team to assure the

quality of the product. The new formed team resembled all the characteristic of a Scrum team.

Company DELTA was the most complex case study because of the size of the teams. Most of the agile methods are recommended for small teams that work on a single project or product. The dynamics of the team can be managed when is smaller than 10 people. In this case study the IT department had 26 people at the beginning of the program. There was a team dealing with business solutions, a frontend development team, a backend development team, two mobile developers, and a quality assurance team. Each group planned their work in iterations based on dependencies with other teams. Management and a small user experience team was providing with requirements. Dependencies were managed between teams by using delayed sprints (e.g. backend planning the work before frontend) but in reality this method was not effective because it created a lot of bottlenecks and an overhead for managing release on the production server. Work in the teams started with a series of interviews with managers and attended planning meetings to understand the team dynamics. Based on the observations workshops on agile methods were designed to introduce people in the agile mindset. Sprint retrospective meetings were started with each team to identify and resolve impediments. After 4 months in the program a reorganization of the department was proposed. Instead of three teams specialized on specific components a cross-functional team model was adopted. The change reduced dependencies and increase autonomy.

Based on experience gained with the teams and involvement in the community a concept for custom agile learning program is proposed. Figure 2 describes the overview of the program.

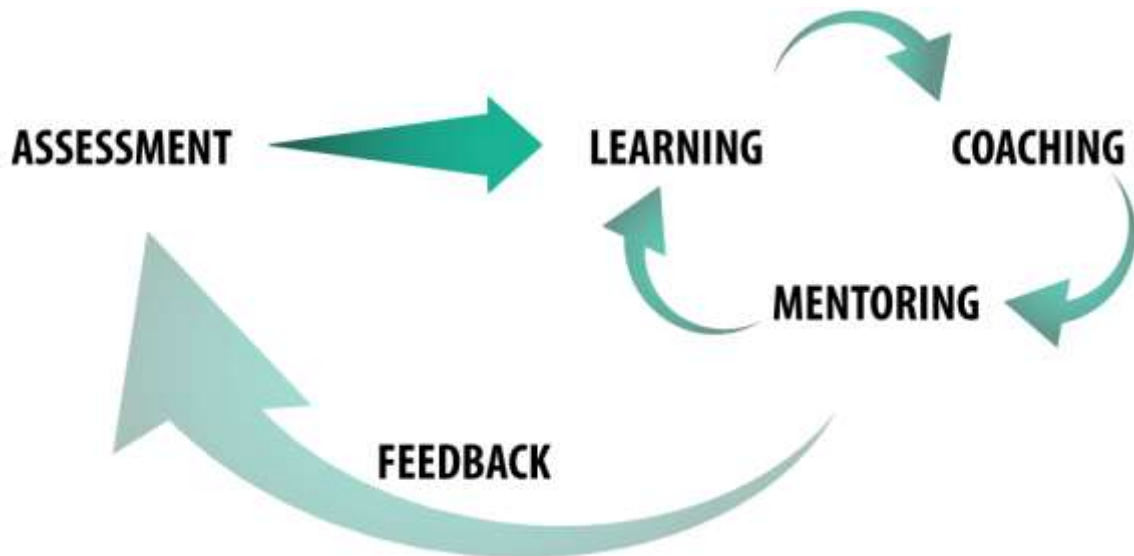


Fig. 2. Agile learning framework

An initial assessment of the process is required to understand the context of the company and identify the goals of the program. Having only theoretical knowledge regarding agile practices is not sufficient to integrate the mindset. A collaborative approach to learning is proposed. Rather than having a training on a specific agile method a custom learning program is designed with the participation of all the stakeholders. In the assessment phase semi-structured interviews and focus groups are done to gather data regarding current practices. The tools and current process is analyzed. Specific goals for the transformation are decided and implemented in an agile manner. By creating a learning environment based on continuous improvement, coaching, mentoring and feedback integration of an agile mindset is more likely to succeed. Future work proposes development of the learning framework by creating an interactive knowledge base on agile practices that is accessible online and can be customized based on client's needs.

4. Discussion

Having an overview of agile practices in the entire software community is important to create a realistic learning and development program. Adopting a new practice in a team involves a significant effort because of the time invested in acquiring know-how and getting all teams members aligned. It's important to

assess what would be the best practices that will increase the overall performance of the process. A change management strategy increases the chances of success. Usually teams start with the meetings prescribed by the Scrum framework but because they lack proper facilitation are more likely to fail in creating the collaborative environment. Due to the nature of project based work there is a focus towards project management practices. This increases the efficiency of the development process but if facing with time constraints and budget constraints the effects on quality are seen on the long run.

The evolution of the community is contributing to the spread of best practices. This is consistent with agile values that emphasize the importance of individuals and interactions in producing quality software. By the use of online social networks we assist to a new way learning that is more decentralized and less dependent on a single source of information. Groups of interests are self-organizing by using online platforms to find people with similar concerns. This could increase the contribution to open source projects and play a significant role in increasing the learning experience to create a culture of technical excellence.

As an Agile Coach, the role involved working hands-on with the teams to help them produce change in the habits and routines by adapting agile practices to their specific context. The

main responsibilities involved designing workshops, facilitating team meetings, and coaching people to a better understanding of agile principles.

Each organization has its own adaptation depending on the contexts. Some organization that rely on Scrum use the term Scrum Master and others that try to be agnostic regarding specific method call it Process Master. In our country the role is yet in development as many organization relay on more on project management having only one person that serves both the team and follows the project goals. In Agile Software Development this is considered a conflicting situation so it proposes that two people share this concerns (e.g. Scrum Master and Product Owner in the Scrum framework).

An agile learning program must address the human characteristics of software development. Working in software teams requires complex problem solving abilities, critical thinking, creativity and teamwork. The ability to coordinate with others in order to achieve common goals is crucial to project success. Involving all the team members in the development of agile practices will reduce resistance to change that affects any transition to a new way of working.

Several characteristics of the Romanian ICT sector are presented in relation with agile best practices. Knowing the big picture aligns expectations when designing an agile learning and development program within a company. Factors that influence the development of software methodologies are discussed.

a) Practice and tools

Developers are more likely to identify with a specific tool or technology rather than with the practice of software development. Most of the jobs are in some extent related with a specific programming language (e.g. C++ programmer, PHP developer, .NET engineer, Java developer). This leads to a high degree of specialization and less focus on business domain knowledge that is important to produce solutions that would satisfy customer needs. Product oriented companies might face challenges to change the mindset towards problem under-

standing and innovative thinking. Agile methodology promotes having a blend between technical knowledge and business knowledge and having an understanding of the overall development process in order to be effective. Ability to build relationships, collaborate and communicate within a team is important to achieve results.

b) Enterprise and open-source solutions

Presence of multinational corporate companies steers the focus towards learning enterprise solutions. Having the advantage to offer high payed jobs even right after finishing college make them attractive for junior developers. Universities respond to companies needs by creating courses and ensure the right skills required to get a job as soon you finish college. Companies develop body of knowledge and provide certification programs.

Use of open-source is often associated with startups and small companies as they require more flexible solutions to enhance business agility and lack the funds to implement enterprise software. Learning an open-source tools is more community oriented and is done by joining mail groups, online forums, attending informal events and working freelance jobs.

c) Public and private sector

There is a general lack of transparency regarding acquisition of software and IT services in the public sector. Contracts are allocated to companies that seem to know the intricate of the system and are able to respond with comprehensive documentation, but often fail to provide working software. Private sector is more likely to adopt agile methods as they depend on the ability to deliver and satisfy client needs. Still there is a challenge to adapt practices in the outsourcing context due to contract negotiation. Upfront requirements must be provided. Planning and managing changes are not that flexible. This leads to a more process oriented methodology. Providing only the workforce for specific projects limits the value created.

d) Project and product management

Most of the software initiatives are project oriented due to the focus towards outsourcing and providing services. A great amount of ef-

fort is spent with writing requirements, planning, allocating and tracking activities. Project management is more oriented on deliverables where product management is more oriented on outcomes and customer satisfaction. Agile methodology promotes small projects that lead to a new enhanced increment of the product. Switching from project to product management takes a change in mindset. Building a great product is a challenge to combine multiple disciplines and have people with different skills working together.

e) Technical excellence

Despite the fact that ICT sector is promoted as having great specialists we don't contribute much for the development of the industry worldwide in terms of open source projects or successful products. Only few products that are more likely to be the exceptions that confirm the rule have crossed the borders and made a significant impact. Romanian software community is in a phase of development as most of the companies have sufficient funds now to make investments in learning, research and development. Most of the technologies like programming languages, software frameworks or software platforms are imported and the main focus is on building applications. Engineering best practices like test driven development (TDD), test automation, continuous integration and measuring software quality metrics scored lower in the survey.

f) Education and continuous learning

Education is an important factor in the development of software development practices. Major ICT hubs in Romania are based around universities with tradition in forming well prepared individuals. Companies are starting to increase their involvement with the academic community in order to share best practices and provide with case studies. Developing agile methods takes deliberate practice and requires an attitude of continuous learning.

5. Conclusions

The Romanian ICT sector is getting aligned in terms of methodology and technology with current standards in software delivery. Education and continuous learning play a significant role in the development of best practices. Still

low investments in research and development keeps the productivity below its full potential. Agile Software Development (ASD) brings a new approach in delivering software by taking in account the human characteristics and work relationships.

In order to increase productivity a learning method that relies on agility, collaboration and continuous improvement is proposed. Adapting the methodology to the specifics of the Romanian ICT sector and the Romanian culture is an important factor to gain the desired benefits. Future work proposes development of the learning framework by creating an interactive knowledge base on agile practices that is accessible online and can be customized based on client's needs. Emerging roles in agile teams will be analyzed and an investigation of performance metrics is proposed. The focus of future research will be towards practices that enhance performance and increase the likeliness of creating software products that satisfy customer needs. New approaches to software development are required to accomplish the transition from an industry oriented to outsourcing services to creating products with a higher added value.

Acknowledgement

Preliminary results of the study were presented at "The 14th International Conference on Informatics in Economy" [12] and "The 16th International Conference on Informatics in Economy" [14]

References

- [1] Version One, *The 10th Annual State of Agile Survey*, Version One, 2016, available at www.stateofagile.com
- [2] K. Schwaber, & J. Sutherland, *The Definitive Guide to Scrum: The Rules of the Game*, 2016, available at <http://www.scrumguides.org>
- [3] K. Beck, *Extreme programming explained: embrace change*. Addison-Wesley Professional. pg. 190, 2000
- [4] J.D. Anderson, *Kanban: Successful Evolutionary Change for Your Technology Business*, Blue Hole Press, pg. 278, 2010
- [5] D. Leffingwell, *SAFe® 4.0 Reference*

- Guide: Scaled Agile Framework® for Lean Software and Systems Engineering*, Addison-Wesley Professional, pg. 576, 2016
- [6] C. Larman, B. Vodde, “*Large-Scale Scrum: More with LeSS*”, Addison-Wesley Professional, pg. 368, 2016
- [7] A.S. Campanelli, F.S. Parreiras, “*Agile methods tailoring – A systematic literature review*”, *Journal of Systems and Software*, vol. 110, pp. 85-100, 2015
- [8] K. Dikert, M. Paasivaara, C. Lassenius, “*Challenges and success factors for large-scale agile transformations: A systematic literature review*”, *Journal of Systems and Software*, vol. 119, pp. 87-108, 2016
- [9] M. Stoica, B. Ghilic-Micu, M. Mircea, C. Uscatu, “*Analyzing Agile Development – from Waterfall Style to Scrumban*”. *Informatica Economică*, vol. 20, no. 4, pp. 5-14, 2016
- [10] Scrum Alliance, “*2015 State of Scrum Report*”, 2016, available at <https://www.scrumalliance.org/why-scrum/state-of-scrum-report/2015-state-of-scrum>
- [11] S. Ambler, “*2014 Agile Adoption Survey Results*”, 2014, available at <http://www.ambysoft.com/surveys/agileJanuary2014.html>
- [12] E.N. Budacu, C.N. Bodea, S. Stancu, “*Agility in The IT Services Sector: A Study for Romania*”, *Proceedings of the IE 2015 International Conference*, Bucharest, 2015
- [13] M.K. Spayd, L. Adkins, “*Developing Great Agile Coaches: Towards a Framework of Agile Coaching Competency – Part I*”, 2011
- [14] E.N. Budacu, “*A Study on Agile Software Development Community of Practice in Bucharest*”. *Proceedings of the IE 2017 International Conference*, (pg. 358 - 363). Bucharest, 2017.



Eduard Nicolae BUDACU has graduated the Faculty of Cybernetics, Statistics and Economic Informatics from the Bucharest University of Economic Studies in 2010. He has graduated the SIMPRE - ERP oriented Master's Program from the Bucharest University of Economic Studies in 2012. He is currently a PhD Student at the Economic Informatics Doctoral School studying the topic of agile methodologies for developing mobile solutions. His main field of interest is agile software development. He works as an agile coach to

helps software development teams produce positive change in order to achieve performance using agile principles and practices. He works with companies to define learning and development strategies for agile transformation.