Scheme of Global Scrum Management Software

M. Rizwan Jameel Qureshi, Isaac Sayid
Department of Information Technology, Faculty of Computing and Information Technology, King Abdul Aziz
University, Jeddah, Saudi Arabia
Email: anriz@hotmail.com, Isaac.sayid@gmail.com

Abstract—Many software development companies faced the challenge of coupling agile models and global software development in distributed projects. The challenges are related to the communication, management and control of the development process. These challenges emerged because global development involves developers from different geographical locations, time zones and with different cultures. Coupling agile models and global software development seems impossible at first glance, as they require frequent communications, rapid development, and resources management of distributed teams. However, researchers proposed several solutions and advices included tailoring of agile practices and addition of non-agile practices. Nevertheless, further efforts were needed to activate and unify these solutions. This paper introduces a new web based project management system for Scrum projects called global scrum management. It is a web application that manages Scrum process from planning phase to delivery of the last product increment. Moreover, the application features social networks functionalities to provide seamless communication, collaboration and knowledge transferring to the distributed team members. Also, developers' actions to the sprint tasks, including updating task status, will be reflected to the burn down charts of the sprint and product backlog instantly. Several diagrams are provided in this paper to explain the solution including UML diagrams.

Index Terms—Scrum, global development, agile, social scrum, distributed development.

I. Introduction

Global software development (GSD) has become widespread in the software development market. GSD includes outsourcing, offshore teams and partnerships. There are several benefits from GSD. The most important one is that the costs of the development are reduced. Also, GSD provide access to better-skilled developers around the world and lead to follow the best known practices. In the contrary, GSD introduces several challenges related to the communication, management and control of the development process. These emerge because GSD involves distances in three major dimensions: geographical, temporal and socio-cultural [1]. As a result, many researchers proposed a number of solutions to minimize the side effects of GSD, e.g. encouraging communications between members and build an environment for knowledge transferring [2].

Agile software development is introduced to overcome several issues, e.g. projects with high level of uncertainty. Agile is a set of methods based on incremental and iterative development [3]. According to agile manifesto, the most efficient and effective method of conveying information to and within a development team is face-to-face conversation. Also, acceptance of changes, motivating individuals and trusting them will all help to get the job done.

However, applying agile methodology to GSD environments is not straightforward, as they require continuous face-to-face communication. Consequently, to get agile methods advantages in GSD, the agile practices should be tailored in a way that ensures a proper application of agile methods implicitly.

Scrum is an agile development framework in which cross-functional teams develop products or projects in an iterative, incremental manner [4]. It divides development in cycles of work called Sprints. At the end of each Sprint, the team reviews the Sprint with the product owner and other stakeholders, and demonstrates what it has been built. People receive feedback that can be added in the next Sprint. Scrum emphasizes working product at the end of the Sprint that is really "done". Scrum seems to work extremely well as an agile project management approach for global software development [5]. A number of solutions and advices are provided on how to tailor the agile practices and Scrum artifacts in GSD. However, there was less focus on software systems or tools that could be used to achieve the literature solutions or advices. Also, we know of no online Scrum management software with the goal of tracing employee availability, performance, and centralized resource management and monitoring. This paper introduces new project management software that will address communication, development and management issues when applying Scrum in GSD.

The paper is structured as follows. First, we introduce challenges of using agile practices, especially scrum practices, and then we present suggested solutions in GSD projects. Section 3 describes the research problem. A global scrum management (GSM) web application is introduced as a solution to research problem in section 4.

II. Related Work

A number of solutions have been proposed for the common mistakes while managing distributed agile teams [6]. The solutions are described as lessons learned while leading workshops. The mistakes included using Scrum as it is without changes and ignoring the time zone differences of distributed teams. Following are the solutions proposed for the mistakes.

1. Use scaled agile models with adaption to fit your needs
2. Select developers who are close in time zones.
3. Overcome misunderstanding and increase the trust between team members by allowing them to travel to meet each other and discuss about their cultures.

However, these solutions were unclear in terms of implementation. In other words, nothing describes the exact changes or adaptions on Scrum model.

Dorairaj et al. described different ways used to build trust and improve communication between distributed team members [7]. Examples included social communication, face to face interaction, sharing of knowledge through presentations or by answering asked questions, initiative collaboration, and willingness to learn from mistakes. Nevertheless, further efforts are needed to develop or suggest software tools to manage the teams and enhance knowledge transfer and communication between team members.

Scrum seems to work extremely well as an agile project management approach [5]. To prove this, Pries-Heje et al. carried out a longitudinal case study of a distributed project using Scrum across Denmark and India. They used three selected theoretical frameworks on the analyzed data. Conclusion showed that Scrum works so well because it provides communication, social integration, control, and coordination mechanisms that are especially useful for distributed and agile project management. However, further efforts are needed to develop or suggest online agile project management tools to manage the teams and enhance knowledge transfer and communication between team members.

Project development progress is exhibited to changes due to variety of factors. However, in distributed projects, maintaining these changes by the team members become harder. This will lead to inaccurate work progress [8]. Alyahya et al. proposed a solution that supports the effective management of development progress by tightly integrating task or story progress data with the versioning data of development to keep progress consistency within team members. However, the implementation in real environment needs to be discussed and agile project management software need to be developed to help following up the work progress through single source by different members.

Identifying communication issues would help researchers to acknowledge different problems that could occur in agile global software development (AGSD) [9]. Kamaruddin et al. used literature survey to gather needed information. The authors discussed a variety of issues in GSD. These issues included different culture and language of the team members, lack of direct communication (face to face), different working hours/time zones, and unprepared communication Tools. Nevertheless, there are tools that can be suggested or developed to overcome many communication issues described previously. A web application for agile project management would enhance communication and knowledge transfer between team members.

Scrum methodologies should be tailored to meet the different needs of distributed projects [10]. Bass, J.M. used a grounded theory approach to identify nine different roles of the product owner. They are: groom, task prioritize, release master, technical architect, governor, communicator, traveler, intermediary and risk assessor. Tailoring Scrum methodology to the distributed projects is explored from the practitioners' descriptions to the previous roles. The author focused mainly on tailoring the product owner role in distributed environment. However, further investigation is needed to identify how to tailor other Scrum roles like development team and Scrum master.

Jalali et al. reviewed conference papers from 1999–2009 about the use of agile practices on the distributed agile software development [11]. The main purpose was to highlight which circumstances of agile methodologies, in which global software settings would be applied efficiently. The analysis and results revealed that agile practices and methodologies were mostly tailored to become applicable. Also, there was no systematic review that covers practicing all agile methods with all possible GSE settings. Further efforts are needed to provide detailed steps on how to tailor the agile methodology to the practitioners.

Estler et al. [12] have examined a total of 66 industry projects. The study compared agile (Scrum, XP, etc.) vs. structured (RUP, waterfall) processes to determine the choice impact. They evaluated the correlation between process type and different measures like success, importance, team motivation, and real-time communication. The results have shown that selecting an agile rather than a structured process does not appear to be a crucial decision for globally distributed projects. However, researchers need to investigate more to identify which agile practices are common and followed in practice.

Noordeloos et al. [13] conducted a qualitative data analysis approach and found that moving from RUP to Scrum brings a positive effect in requirements engineering, communication, cost management and cross-functionality of the distributed teams. They also observed a negative change regarding the development pace and delivery time due to the lack of documentation and missing the tracing lines. As a future work, more investigations are needed on how developers communicate with each other in daily Scrum meetings and how they transfer the knowledge between each other.

Ghosh [14] listed the main challenges when implementing distributed Scrum on the projects. For example, communication factors like: distance, culture, time zone, and language. Ozawa et al. [2] were able to

overcome many differences by encouraging communications between members. They have not only adopted Scrum, but also built an environment for knowledge transferring and communication between members. Instead of making documents clearer to the offshore team, they have allowed for communication to understand each other. However, some differences between team members are only solved with the time and project advancements.

Table 1. Limitations of Related Work

Title	Limitations
Lessons Learned from Leading Workshops about Geographically Distributed Agile Teams [6]	• The solutions provided for adapting the agile methodology to overcome non-collocated team members is unclear in terms of implementation. • Finally, virtual meetings and knowledge transferring are always necessary but there were less focus on the tools that could be used to achieve this goal easily to enhance the knowledge transfer and communication.
Agile Software Development with Distributed Teams: Agility, Distribution and Trust [7]	• The author concentrated mainly on how to build trust between team members. However, keeping this trust and avoid declining it is also important. • Further efforts are needed to develop or suggest software tools to manage the teams and enhance knowledge transfer and communication between team members.
Why Scrum works: A case study from an agile distributed project in Denmark and India [5]	• The research was focused on two different scrum teams in two different locations where communication is done mostly by the Scrum master of each team and the project manager. What about one distributed Scrum team with one scrum master? • Moreover, existing of a project manager role violates the Scrum methodology because it has no such a role.
Co-ordination Support for Managing Progress of Distributed Agile Projects [8]	• The approach proposed by the author was clear enough. However, the implementation in real environment needs to be discussed. • Further efforts are needed to develop web-based agile project management software to help following-up the work progress in different levels.
Chaos issues on communication in Agile Global Software Development [9]	• There are tools that can be suggested or developed to overcome many communication issues described in the paper. A web application for agile project management would enhance communication and knowledge transfer between team members.
Agile Method Tailoring in Distributed Enterprises: Product Owner Teams [10]	• However further investigation is needed to identify other Scrum roles tailoring like development team and Scrum master.
Agile Practices in Global Software Engineering – A Systematic Map [11].	• Further research would help to study these changes and provide steps to the practitioners on how to tailor the agile methodology.
Agile vs. Structured Distributed Software Development: A Case Study [12]	• As a future work, researchers are planning to investigate more agile and structured projects, to identify which agile practices are common and followed in practice.
From RUP to Scrum in Global Software Development: A Case Study [13]	• Furthermore, researchers need to explain how RUP roles are matched with Scrum roles after the transition. In other words, what were the duties of the product owner before and after transition?
Challenges in Distributed Scrum [14]	• What technologies should they adopt to manage online team communication, documents and work progress?
Adapting Agile Methodology to Overcome Social Differences in Project Members (Japanese and China) [2]	• Adopting agile practices was not enough to succeed in an offshoring project.
Agile requirements prioritization in large-scale outsourced system projects: An empirical study [15]	• A web application for agile project management would enhance communication and quicken sharing of prioritized requirements between team members. Also, it can add central source of data and instant updates to the document and prevent data inconsistency.
Influences on Agile Practice Tailoring in Enterprise Software Development [16]	• Further exploration is required to understand the lack of awareness or lack of commitment by development teams to XP practices.
Distributed agile: project management in a global environment [17]	• Techniques need to be suggested for enhancing communication between team members.
Using Scrum in a Globally Distributed Project: A Case Study [18]	• Poor communication among distributed meetings due to cultural and geographical distance.

Daneva et al. [15] used interviews as a data collection method and grounded theory for data analysis. The research results were the following:

- Requirements dependency is the key factor toward successful agile implementation in large distributed projects.
- Risk is the most important prioritization criterion in the setting of large distributed agile projects
- "Delivery stories", a new terminology added a good value to the agile development. It is the "user stories" with addition of technical implications, effort estimation and associated risk.

However, researchers did not provide a clear description of how vendors deal with agile requirements engineering in distributed and outsourcing contexts.

Bass [16] conducted a study to find out the reasons for tailoring agile practices within the context of large enterprises (distributed) projects. The author performed interviews with 19 practitioners. The final results included the following reasons for tailoring agile practices:

- Scum roles and practices did not conflict with global projects policies but agile practices from the XP tradition were not so widely adopted.
- Test driven development did not integrate well within global projects where independent quality assurance teams were constituted as separate departments
- Continuous integration was found to be challenging due to time and regressions testing requirements of the global software products

Seiyoung et al. [17] addressed many successful practices in My Yahoo! by the global product team. They are face-to-face communication, self-organizing teams, collaboration, changing requirements, frequent delivery, technical excellence and retrospectives. Also, the author suggested a framework for software globalization project management using a distributed agile approach.

Maria et al. [18] performed seven interviews in the development organization distributed between Norway and Malaysia. They described how scrum practices were successfully applied. These practices included using teleconferencing and web cameras for daily scrum meetings, synchronized four-week sprints, and weekly scrum-of-scrums. Also, using of Jira for backlog management. Other non-Scrum practices included frequent visits, unofficial distributed meetings, nightly builds, and automated testing. Table 1 shows the literature review summary of each paper with the limitations.

III. Problem Definition

Applying the agile software development with distributed teams is not straightforward because of the different time zones, cultures, and geographical locations.

Since agile development largely relies on daily communication, acceptance of changes, rapid development and knowledge transfer according to agile manifesto, agile practices should be adapted to align with the global software development. The research question is as follows [5-9][11][2][15][17].

How to improve communication, development, transferring of knowledge and resources management of distributed Scrum agile development methodology?

IV. The Proposed Solution

A global Scrum management (GSM) is a web application that manages Scrum agile development roles, events, and Scrum artifacts, from planning to closure. The user interface of GSM is inspired from social networks. The application supports a variety set of technologies to provide centralized control accompanied by seamless collaboration and development to the distributed team members. Following is a detailed description of the proposed solution.

A. Management

GSM automates Scrum framework with instantly updated charts and statistics that are drawn based on the project data. GSM web application consists of four user groups: system admin, product owner, development team and Scrum master. System admin can create new project instances, create new Scrum teams, assign Scrum team to a project, assign users to a Scrum team and define the user group of each user. A product owner can write user stories, prioritize tasks and then add them to the product backlog. He/she also has access to the project evaluation tools such as the burn down chart. Burn down chart for Scrum iteration/sprint is useful for predicting the end time of completing all the tasks and efforts. Burndown chart is depicted in Fig. 1. Other statistics may include the overall progress and performance of a project, a Scrum team or a single user. These instant statistics and charts may help the product owner along with the development team to take corrective and future decisions. Moreover, this high level of monitoring and tracking that is provided using the charts and statistics can encourage the development team to be more self-organized. Thus, GSM achieves Scrum development team definition.

A user with Scrum master role is responsible for documenting and facilitating the Scrum events. The Scrum master chairs key events. Scrum events include Sprint planning meeting, daily Scrum meeting, sprint review meeting and sprint retrospective meeting [19]. The communication tools and the documents related to each event are discussed next.

B. Communication

GSM embraces effective collaboration within distributed/virtual Scrum teams. Distributed Scrum projects are socialized by providing social like web environment to the team members. The social web environment improves collaboration, problem solving, sharing of digital stuff, archiving project documents,

knowledge transferring and many other activities that will together provide significant communication value to the distributed team members. Fig. 2 demonstrates the way how GSM works. On the contrary, Fig. 3 depicts global Scrum development without GSM. Further, central source of data with instant updates will be available at any time to the project members. GSM's social environment works as the following: sprint instance is created with each planning meeting. Each sprint instance
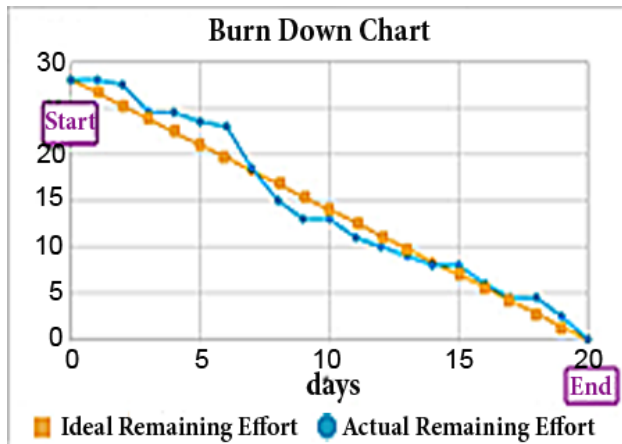

Fig. 1. Burn Down Chart

has its own data items. Data items of a sprint may include a discussion board, sprint backlog, goals of the sprint, documents and burn down chart. During sprint period, sprint backlog is updated regularly by the development team members. Related topics and meetings are discussed in the discussion board and knowledge is transferred between team members surpassing the time zones differences. Sprint related documents and files are uploaded or updated providing central access to the project resources. In addition, the web application has its own chat and instant messaging software for communication between Scrum team members. Also, services like Skype or Hangout may be used for face-to-face meetings between project members if needed. Members can post answers to the daily meetings questions in discussion boards instead of conducting videoconference meetings daily to avoid issues of time zones and to save the time. Meeting outlines and conclusions are denoted by the Scrum master for later referencing. At the end of sprint iteration, learned lessons and other improvement suggestions are added to the sprint instance.

C.   Development

GSM include bugs tracking and code revisions services which can help the development team to manage the work. GSM provides development technologies either built in or integrated with other third-party cloud solutions such as Github or Jira. Whatever the choice, these services keep track of reported software bugs and store older versions of the software providing consistent copies of the code to the developers. The developers of GSM application will be assigned tasks selected from

sprint backlog by the product owner. Also, developers can assign tasks to themselves, update the progress of these tasks and also can mark them as finished. Developer's actions will be directly reflected to the system charts and statistics. Fig. 4 depicts GSM use case diagram. In addition to the previous, GSM application layout adapt to the screen size of different types of devices including smart mobile phones and tablets either in portrait or landscape modes. As a result, quick and portable access to the resources is available to the project members. The application is designed to work in different contexts. These contexts presume that both the product owner and Scrum master are collocated with the client with one or more offshore development teams.
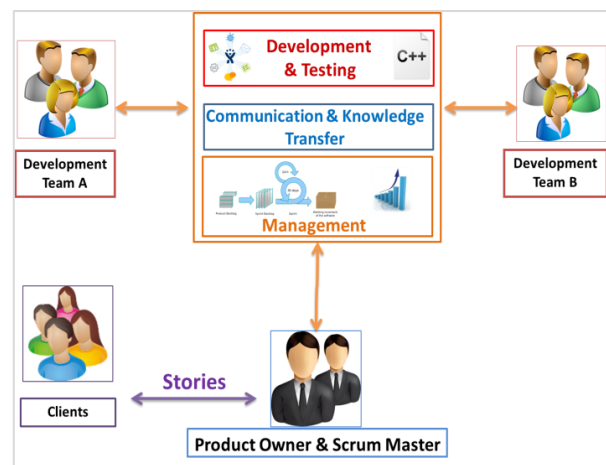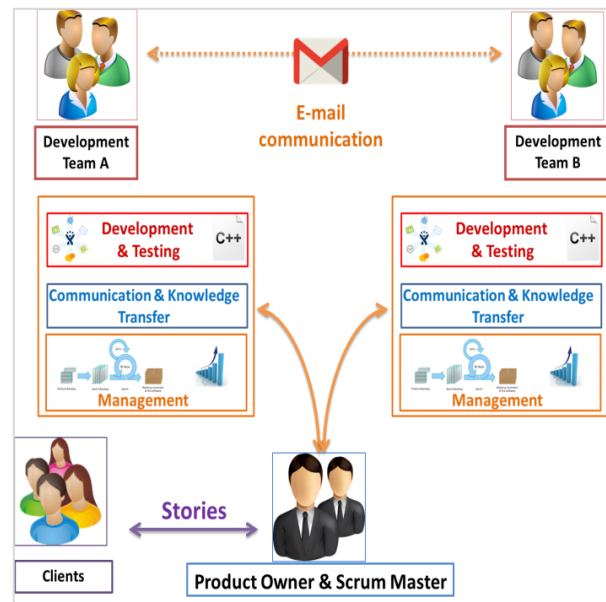

Fig. 2. Global Scrum Management (GSM)

Fig. 3. Normal Scrum Project Management

V.   Conclusions And Future Work

In this paper, we described the new proposed solution for distributed Scrum agile development methodology called global Scrum management (GSM). Also, we

I.J. Information Engineering and Electronic Business, 2015, 2, 1-7

introduced how GSM can be used to improve communication, development, transferring of knowledge and resources management in GSD projects. The overall analysis of the questionnaire showed positive results. Scrum by nature encourages frequent communications between team members which makes it more suitable for distributed projects. GSM provides a social module which keeps communicating of members with each other quickly and easily.

A single high level use case diagram of the new proposed GSM solution is obviously limited. In the future, we plan to add more UML diagrams with more details to explain GSM. Moreover, a working prototype may help in visualizing the system functions and features. Also, it will strengthen the collected results. Finally, GSM implementation will be the key to study the system feasibility in practice. A number of case studies will be required therefore to test the feasibility, success and effectiveness of GSM. We think that GSM may raise the need for similar systems that could function with other distributed agile methodologies like XP. Moreover, GSM is not restricted to GSD, but it can be used to solve communication issues of the collocated members, e.g. in Islamic societies that prevent mixing of males and females in the workplace.
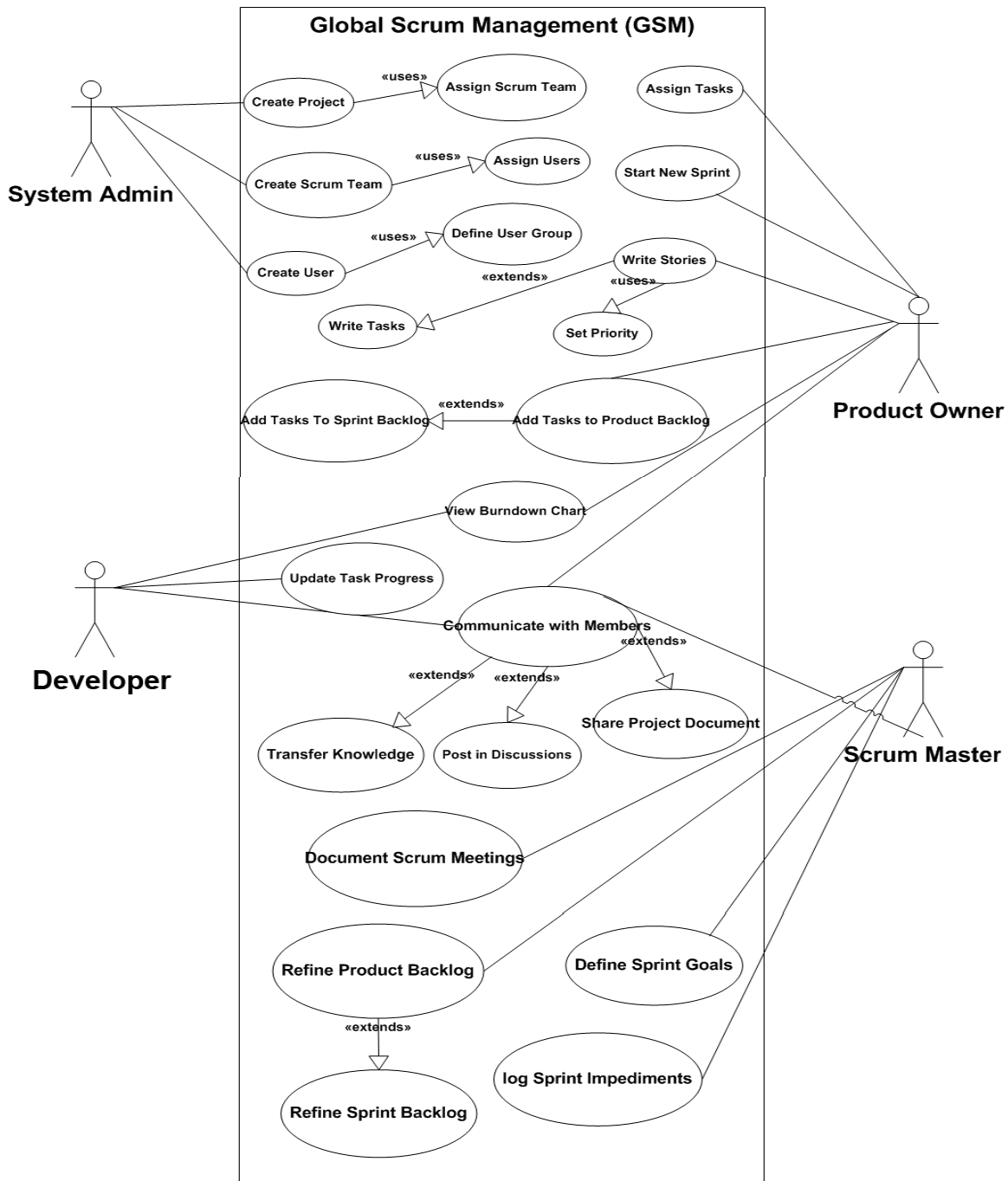
Fig.4. GSM Use Case Diagram

REFERENCES

[1] P. J. Ågerfalk, B. Fitzgerald, H. H. Olsson and E. Ó. Conchúir, "Benefits of global software development: the known and unknown," in *international conference on Making globally distributed software development a success story*, Springer-Verlag Berlin Heidelberg, 2008.

[2] H. Ozawa and L. Zhang, "Adapting Agile Methodology to Overcome Social Differences in Project Members," in *Agile Conference (AGILE), Nashville*, TN, 2013.

[3] "Principles behind the Agile Manifesto," The Agile Alliance, 11 2 2001. [Online]. Available: http://agilemanifesto.org/principles.html. [Accessed 29 5 2014].

[4] P. Deemer, G. Benefield, C. Larman and B. Vodde, The Scrum Primer: A Lightweight Guide to the Theory and Practice of Scrum V2, InfoQ.com, 2012.

[5] L. P.-H. J. Pries-Heje, "Why Scrum Works: A Case Study from an Agile Distributed Project in Denmark and India," in *Agile Conference (AGILE)*, Salt Lake City, UT, 2011.

[6] J. Rothman and S. Hastie, "Lessons Learned from Leading Workshops about Geographically Distributed Agile Teams," *Software, IEEE*, pp. 7 - 10, 2013.

[7] S. Dorairaj and J. Noble, "Agile Software Development with Distributed Teams: Agility, Distribution and Trust," in *Agile Conference (AGILE)*, Nashville, TN, 2013.

[8] S. Alyahya, W. Ivins and W. Gray, "Co-ordination Support for Managing Progress of Distributed Agile Projects," in *Global Software Engineering Workshop (ICGSEW)*, Helsinki, 2011.

[9] N. Kamaruddin, N. Arshad and A. Mohamed, "Chaos issues on communication in Agile Global Software Development," in *Business Engineering and Industrial Applications Colloquium (BEIAC)*, Kuala Lumpur, 2012.

[10] J. Bass, "Agile Method Tailoring in Distributed Enterprises: Product Owner Teams," in *Global Software Engineering (ICGSE)*, Bari, 2013.

[11] S. Jalali and C. Wohlin, "Agile Practices in Global Software Engineering - A Systematic Map," in *Global Software Engineering (ICGSE)*, Princeton, NJ, 2010.

[12] H.-C. Estler, M. Nordio, C. Furia, B. Meyer and J. Schneider, "Agile vs. Structured Distributed Software Development: A Case Study," in *Global Software Engineering (ICGSE)*, Porto Alegre, 2012.

[13] R. Noordeloos, C. Manteli and H. Van Vliet, "From RUP to Scrum in Global Software Development: A Case Study," in *Global Software Engineering (ICGSE)*, Porto Alegre, 2012.

[14] G. Ghosh, "Challenges in Distributed Scrum," in *Global Software Engineering (ICGSE)*, Porto Alegre, 2012.

[15] M. Daneva, E. V. D. Veen, C. Amrit, S. Ghaisas, K. Sikkel, R. Kumar, N. Ajmeri, U. Ramteerthkar and R. Wieringa, "Agile requirements prioritization in large-scale outsourced system projects: An empirical study," *Journal of Systems and Software*, vol. 86, no. 5, p. 1333–1353, 2013.

[16] J. Bass, "Influences on Agile Practice Tailoring in Enterprise Software Development," in *AGILE India (AGILE INDIA)*, Bengaluru, 2012.

[17] S. Lee and H.-S. Yong, "Distributed agile: project management in a global environment," *Empirical Software Engineering*, vol. 15, no. 2, pp. 204-217, 2010.

[18] M. Paasivaara, S. Durasiewicz and C. Lassenius, "Using scrum in a globally distributed project: a case study," *Software Process: Improvement and Practice*, vol. 13, no. 6, p. 527–544, 2008.

[19] K. Schwaber and J. Sutherland, "Scrum Guide," Scrum.org, Available at: https://www.scrum.org/Portals/0/Documents/Scrum%20G uides/2013/Scrum-Guide.pdf. [Accessed 29/ 5/2014].

[20] U. Nations, World Investment Report 2004 - The Shift Towards Services, New York and Geneva: United Nations, 2004.

Authors' Profiles

Dr. M. Rizwan Jameel Qureshi received his Ph.D. Computer Sciences degree from National College of Business Administration & Economics, Pakistan 2009. He is currently working as an associate professor in Department of Information Technology, Faculty of Computing & Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. He is the best researcher awardees of King Abdul-Aziz University Saudi Arabia in 2013 and Department of Computer Science, COMSATS Institute of Information Technology Pakistan in 2008.

Isaac Sayid was born in 15th September 1989 in KSA, Makkah. He received B.Sc. Degree in Information Technology. Faculty of Computing & Information Technology from King Abdulaziz University, KSA in 2013.

He is currently a graduate student at King Abdulaziz University. While undergraduate student, he worked as a tutor for four semesters with the college programmers' team to teach students advanced java programming language. He also belongs to the development team of complaints reporting system (CRS). CRS was developed as an e-government system that aims to facilitate and speed up the process of reporting a noticed problem to the concerned authority by getting the benefit of smartphones, web technology, tablet devices and GPS service. His role was designing all system interfaces and implementing the website services and other server side components.