

A Gamification approach for Distributed Agile Delivery

Vibhu Saujanya Sharma
Accenture Technology Labs
Accenture, Bangalore, India
vibhu.sharma@accenture.com

Vikrant Kaulgud
Accenture Technology Labs
Accenture, Bangalore, India
vikrant.kaulgud@accenture.com

P. Duraisamy
Accenture Technology Labs
Accenture, Bangalore, India
p.duraisamy@accenture.com

ABSTRACT

Large organizations need to be nimble in delivering software solutions for meeting rapidly changing business requirements and technology landscape. Following Agile principles of software development is a natural choice. However, to truly leverage the power of Agile, big organizations need to be able to utilize distributed teams effectively. Agile relies hugely on shared context and awareness among team members and this can become a stumbling block among such geographically dispersed teams. Moreover, in such large projects there is a need for incentivizing quick delivery of user stories so that the teams have a constructive sense of competition and are recognized in-process. Here, we describe a gamification based approach which promotes quicker completion and acceptance of user stories in such distributed Agile projects. Our approach captures important events from the development environment and then helps create project-wide awareness regarding the progress of different teams. A model of earning revenue for faster delivery of user stories is used to determine the leading team at the end of each sprint. This approach has been implemented in an Agile process guidance and awareness workbench that we are piloting within our organization.

Keywords

Distributed Agile; Agile Gamification; Software Delivery

1. INTRODUCTION

Global businesses and IT services companies have for a long time adopted a global software delivery model wherein software teams are geographically distributed. While cost arbitrage was a motivating factor, today we observe that talent sourcing is a reason for distributed teams as well. Given that software is driving innovations in most businesses and short time-to-market for Digital solutions (mobile apps etc.), continuous software delivery is becoming mainstream. Secondly, to address the changing technology landscape and

rapidly changing customer requirements, software development has to be flexible to accommodate new technologies and customer requirements while remaining agile. There is already a substantial investment in building global software teams and organizations want to retain this global 'team architecture', but adopt Agile methods to address the technical and business imperatives.

Irrespective of the reasons, distributed software delivery teams are becoming common. However, distributed teams create challenges for effective coordination of development tasks [1]. In a waterfall software development model, these coordination issues can be mitigated through appropriate project planning, resource management, and software architecture. However, adoption of Agile methods in a distributed delivery environment is a big challenge [2].

As mentioned in the Agile Manifesto [3], amongst other principles, there is an emphasis on face-to-face interaction between all stakeholders of the agile software project. The physical and social interaction which is considered a key benefit of Agile [4], automatically gears the teams to be aware of mutual progress as well as promotes members to strive for being more engaged and motivated. The distribution of Agile teams however, lowers the situational awareness of teams and distribution and the lack of personal contact reduces the inherent competitiveness that collocation entails.

Moreover, unlike software product teams which may continue to work on a particular project for a very long time, software delivery project teams are much more prone to change. Based on demands and skills, employees may be re-assigned to different teams or projects at relevant points of time. Hence, it becomes much more important for delivery projects to ensure that individuals and teams are rewarded in-process when they achieve important milestones better/faster than others. Yet another aspect of this is in case of distributed teams and individuals, project members may lose perspective of how well (or not) are they performing as compared to their counterparts in other locations. In such a scenario, an in-process approach that rewards best performance and creates awareness of such achievements within distributed teams can be a vehicle to incentivize everyone to strive for more.

To address these challenges, we are working on technology-led solutions for enhancing situational awareness and introducing competition in distributed Agile delivery projects. We adopt a data-driven approach based on data which collects software development and process data through automation or digital process governance tools. Analytics, role-specific visualization, and gamification techniques are

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GAS'16, May 16 2016, Austin, TX, USA

© 2016 ACM. ISBN 978-1-4503-4160-8/16/05...\$15.00

DOI: <http://dx.doi.org/10.1145/2896958.2896966>

used to enhance situational awareness and incentivize software team members. In this paper we present an approach for gamification in the implementation phase of Agile delivery. Our approach lets teams earn ‘revenue’ each time they are able to implement a user story and get it accepted by interacting with the product owner, earlier than the end of sprint. The approach provides an interactive real-time dashboard of such revenue earned by the various teams and helps recognize the winning team at the end of each sprint. The rest of the paper is organized as follows: Section 2 delves into agile delivery with the design and considerations of our gamification approach. Section 3 provides a deep dive in our approach implementation and usage, and section 4 concludes the paper with some of our observations.

2. GAMIFICATION FOR INCENTIVIZING FASTER AGILE DELIVERY

Gamification in software engineering is a relatively recent area research. The survey in [5] presents a picture of the various approaches for gamifying software quality. In [6] the authors present an approach to gamify removal of problematic patterns from code. There is also some work in gamification for enforcing certain compliances [7]. One of the initial and very popular gamification approach in Agile practices is *Planning Poker* [8] which is frequently used to do release planning and story point estimation. But in general, there is relatively little work in gamification of Agile delivery with most of it focusing on code metrics like quality [9] and to a limited extent in documentation and testing [10].

In the context of distributed Agile delivery, one important business imperative is to be able to leverage geographically spread out teams to deliver user stories fast. As part of the sprint planning, at the beginning of each sprint, certain user stories are assigned to different teams and thereon to individuals within those teams. The teams are then supposed to deliver these user stories before the end of that sprint. This involves not only implementing them but also confering with the Product Owner and making sure that (s)he ‘*Accepts*’ the software artifacts corresponding to those user stories.

The teams are assigned a set of user stories with certain *Plan Estimates* which are linked to the complexity and size of the user story and expected implementation and may involve practices like *Planning Poker* [8] or project specific practices. A team’s portfolio of user stories is typically assigned by the project manager in consultation with the team leads and scrum masters based on the team sizes and expertise. Note that the approaches for assignment of Agile user stories, are not within the purview of our work.

For incentivizing speedy delivery of user stories, we introduce a gamification approach that rewards teams that are able to do it before the end of each sprint. Our approach is at a team level because we found that in case of distributed teams working on the same Agile delivery project, user stories are assigned to teams first and then an internal assignment of those happens to individuals. So the need for awareness and competition is *between* teams rather than *within* teams, wherein because of collocation, Agile principles are more naturally followed.

The game begins post the assignment of user stories to the teams. Each user story’s *Plan Estimate* is fixed as the *Revenue* needed to deliver that user story by the end of the

sprint. If a team spends the whole sprint completing the user story, then that user story is assumed to *Cost* the team the whole *Revenue* associated with that user story. However, if the team delivers it before the end of sprint, say in half the duration of the sprint, then the team is accordingly only assumed to have spent half the *Revenue* associated with the user story as *Cost*. In other words,

$$Cost_i = Revenue_i \times \frac{\text{No. of days to deliver the user story } i}{\text{Total no. of days in the Sprint}}$$

The gamification points, which we in this game we call *Net Revenue* for a team w.r.t a particular user story is computed as

$$Net_Revenue_i = Revenue_i + Bonus_i - Cost_i$$

Where *Bonus* is a manually assigned amount that the Project Manager can award a team based on other characteristics of the user story’s implementation like quality, etc.

If however, a team is not able to deliver the user story during the sprint, it is actually penalized by assigning the *Cost* to be twice the *Revenue* for that story. So assuming no bonus is given in such scenarios, the *Net Revenue* assigned for that user story is the *-ve* of the *Revenue* for that story. At any given time, the team’s score is decided by the *Total Net Revenue* for the team which is the sum of *Net Revenues* of all the user stories it is responsible for in the sprint.

$$Team_score = \sum Net_Revenue_i$$

Next we discuss the implementation of this approach.

3. AN IMPLEMENTATION OF OUR GAMIFICATION APPROACH

We decided to build this approach as a part of the pilot version of our in-house distributed Agile delivery workbench, called Agile Workbench. The Agile Workbench is an internal initiative with our Digital practice to address this important area of standardized implementation and governance of the Agile processes. It is being developed to be single-window to quickly bootstrap Agile delivery projects using Agile Distributed Agile specific methods, metrics and dashboards, development tools, and collaboration approaches.

The Figure 1 shows our game implementation approach. The game is based on objective data which is extracted automatically from the project environment. The data regarding the team and process state (like information about the current Release and Sprint) is extracted from the Agile Workbench platform repositories. The data regarding the Agile user story implementation is extracted automatically from the Agile development tools (like Rally [11]) being used for sprint management in the project.

If however, for operational or business reason, a project decides to prefer a manual upload, an output of the sprint management tool can be uploaded as well. Presently, we can process outputs from Rally [11], JIRA [12] and RTC [13] in the pilot implementation. The pre-processor correlates the Agile process and user story data, and populates an internal database with Agile team and activity data. The Gamification engine then utilizes this data to first compute the story costs based on the current state of the stories. The engine uses a set of rules based on the formulation discussed in the last section to compute points (or *Total Net Revenue*) which

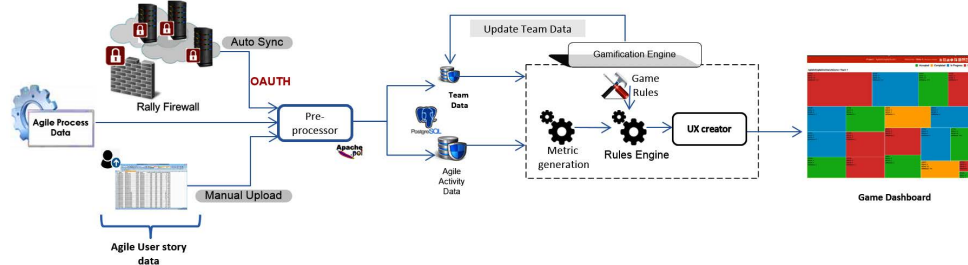


Figure 1: The Agile Delivery Gamification Approach

ID	User Stories	Team	Plan Estimates	Task Estimates	Task Remaining T	Now Accepted	Day Accepted
US1229	UX Account Add	Team-4	13	30	30	Accepted	2
US1232	UX Account Edit	Team-4	0	27	27	Accepted	3
US1233	UX Account Edit	Team-4	13	30	30	Accepted	4
US209	USBR, 24 - Call	Team-4	105	301	301	Accepted	5
US208	UX Order History	Team-4	13	30	30	Accepted	0
US1168	UX Order History	Team-4	0	25	25	Accepted	0
US1145	UX - UX Customer	Team-4	0	20	20	Accepted	0
US1135	Execute MMS In	Team-5 (Integration)	3	8	8	Accepted	0
US909	Billing Portal - Fix	Team-6 (Billing)	32	115	115	Accepted	0
US948	Billing Portal - Mts	Team-6 (Billing)	32	129	129	Accepted	0
US1100	PROCMO - Create	Team-3	13	65	65	Accepted	0
US1065	Menu Enhancements	Team-5 (Integration)	5	32	32	Accepted	0
US1063	Menu Enhancements	Team-5 (Integration)	0	32	32	Completed	0
US1056	Shipping - Edit In	Team-2	247	590	590	Completed	3
US1019	UX Checkout Est	Team-2	21	76	76	Completed	2
US1059	UX Checkout Est	Team-2	5	13	13	Completed	1
US303	195 - Shipping --	Team-2	84	300	300	Completed	0
US1005	195 - Shipping --	Team-2	0	32	32	Completed	2

Figure 2: Team and User Story status

each team has earned. This total net revenue and the hierarchical breakdown of the same is used to create an interactive visualization which is rendered in the Workbench.

One of the simpler visualizations is essentially a tabular view for the project members to see the progress of other teams and the current status of their user stories. This is shown in Figure 2. Note that here the status ‘Accepted’ means that the user story’s implementation is complete and it has been accepted by the Product Owner as well, and therefore is considered to be delivered.

A ‘superuser’ game screen is available to the delivery lead and project manager roles of the project to be able to govern the game. Currently this is where (s)he can gauge at the game’s progress and assign bonus to certain teams’ user stories based on factors not captured in the game. This is shown in Figure 4.

The most intuitive representation of the game however is through an interactive dashboard, as shown in Figure 5. In this treemap dashboard view [14] each team is represented by a rectangle whose area corresponds to the total net revenue earned by that team. Further, each team rectangle is clickable and allows a view of the state of all the user stories of that particular team.

Upon clicking onto any team, one can know more details about the particular team by drilling down into to find other fine-grained details including all the user stories taken up by the team, the status (Defined, In-progress, Completed, Accepted) as well as the details of cost and revenue per user story as shown in Figure 3. One can further drill down into a particular user story to access more details. This besides

serving as a gamification platform also serves as a quick view into the Agile project’s progress and thus helpful to create situational awareness among distributed teams.

4. CONCLUSION

In this paper, we presented the rationale and approach for introducing gamification in Agile software development to address concerns in distributed Agile teams. The approach uses process and software development data as the input to the gamification. Using both data allows us to create an objective view of the project which then forms and input to our game. We also allow for manually enter some of the data - the flexibility is important since project managers have different imperatives depending on the client context, deployment of the environment, or the state of a project.

This gamification approach is being introduced to real-life projects through an Agile Workbench, a digitalized process governance tool that we are building and piloting. We have started conducting experiments in projects to understand the behavioral impact of gamification on distributed Agile teams. Early results are encouraging. We have seen that gamification encourages teams to ensure that their user stories are delivered as soon as possible within respective sprints. A comparison with the sprint velocity of previous, pre-gamification sprints has shown an increase of about 4% in one of the early pilots.

While our approach can in principle be used in non-Agile projects as well, the game lends itself more naturally to Agile because of short sprints which make this game more engaging and allows teams another chance to start afresh with every new sprint every few weeks. Moreover recognitions are recurrent and in-process as opposed to what might happen if this is employed in a waterfall approach. Finally, this game helps bridge the gap created due to non-collocation of teams by becoming a vehicle for awareness and therefore is helpful to establish a shared context between teams (besides competition) which is a cornerstone of Agile.

In the future, we see our work progress in three dimensions. First, we want to continue the experiments in projects to measure the benefits of gamification. We want to use delivery metrics in conjunction with interviews with different roles to gauge the qualitative and quantitative benefits of gamification and different types of games. Second, we want to incorporate more context development data, metrics, location, competency levels etc. in the gamification approach to make it more relevant to project managers. Third, we

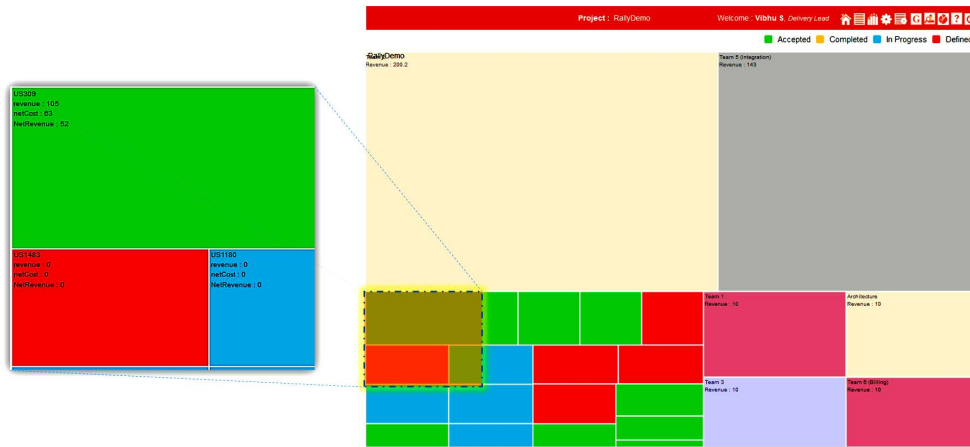


Figure 3: Drill down into the Gamification Dashboard (User-Stories and details by Team)

ID	User Stories	Team	Plan Estima.	Net Cost	Revenue	Total Net B.	Bonus	Day Accepted	New Accepted	Agile Status
US1230	UI - Account	Team 4	13	2.6	13	20.4	10	1	Accepted	None
US1231	UI - Account	Team 4	13	5.2	13	17.8	10	2	Accepted	None
US1232	UI - Account	Team 4	8	4.8	8	13.2	10	3	Accepted	None
US1233	UI - Account	Team 4	13	10.4	13	12.6	10	4	Accepted	None
US309	US991_74	Team 4	105	63	105	52	10	3	Accepted	None
US526	UI - Order Hs	Team 4	13	0	0	0	0	0	Accepted	None
US1188	UI - Order Hs	Team 4	8	0	0	0	0	0	Accepted	None
US1145	UI - UI - Card	Team 4	8	0	0	0	0	0	Accepted	None
US1135	Exercise-MML	Team 5 (Integr)	3	0	0	0	0	0	Accepted	None
US660	Billing Portal	Team 5 (Billing)	32	0	0	0	0	0	Accepted	None
US649	Billing Portal	Team 5 (Billing)	32	0	0	0	0	0	Accepted	None
US1100	PRIMO - Cr.	Team 3	13	0	0	0	0	0	Accepted	None
US1065	Menu Enhanc	Team 5 (Integr)	5	0	0	0	0	0	Accepted	None
US1063	Menu Enhanc	Team 5 (Integr)	8	0	0	0	0	0	Accepted	None
US1056	Shipping	Team 2	247	148.2	247	118.8	20	3	Completed	None
US1191	UI - Checkout	Team 2	21	8.4	21	22.6	10	2	Completed	None
US1059	UI - Checkout	Team 2	5	1	5	14	10	1	Completed	None
US553	196 - Shippi	Team 2	84	0	0	0	0	0	Completed	None
US1052	196 - Shippi	Team 2	8	3.2	8	14.8	30	2	Completed	None
US1054	196 - Shippi	Team 2	8	0	0	0	0	0	Completed	None

Figure 4: SuperUser Interface

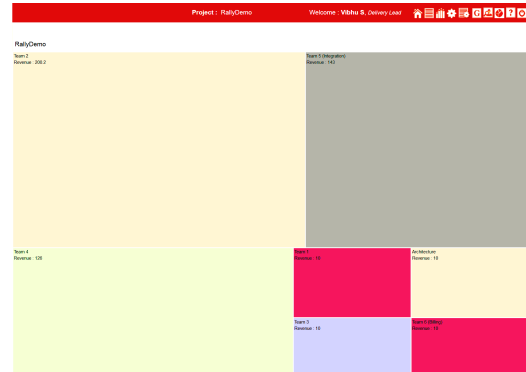


Figure 5: Gamification Dashboard showing Teams standings

are exploring how team members can be guided with contextual information to help them improve adoption of best practices.

5. REFERENCES

- [1] J. D. Herbsleb and D. Moitra, "Global software development," *Software, IEEE*, vol. 18, no. 2, pp. 16–20, 2001.
- [2] B. Ramesh, L. Cao, K. Mohan, and P. Xu, "Can distributed software development be agile?" *Communications of the ACM*, vol. 49, no. 10, pp. 41–46, 2006.
- [3] J. Highsmith and M. Fowler, "The agile manifesto," *Software Development Magazine*, vol. 9, no. 8, pp. 29–30, 2001.
- [4] E. Whitworth and R. Biddle, "The social nature of agile teams," in *Agile Conference (AGILE), 2007*, Aug 2007, pp. 26–36.
- [5] J. Vargas-Enriquez, L. Garcia-Mundo, M. Genero, and M. Piattini, "A systematic mapping study on gamified software quality," in *Games and Virtual Worlds for Serious Applications (VS-Games), 2015 7th International Conference on*, Sept 2015, pp. 1–8.
- [6] S. Arai, K. Sakamoto, H. Washizaki, and Y. Fukazawa, "A gamified tool for motivating developers to remove warnings of bug pattern tools," in *Empirical Software Engineering in Practice (IWESEP), 2014 6th International Workshop on*, Nov 2014, pp. 37–42.
- [7] C. R. Prause and M. Jarke, "Gamification for enforcing coding conventions," in *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, ser. ESEC/FSE 2015. New York, NY, USA: ACM, 2015, pp. 649–660.
- [8] J. Grenning, "Planning poker or how to avoid analysis paralysis while release planning," 2002.
- [9] C. R. Prause, J. Nonnen, and M. Vinkovits, "A field experiment on gamification of code quality in agile development."
- [10] "Agile leagues - agilegamification.org," <http://www.agilegamification.org/agile-leagues/>.
- [11] H. Smits and G. Pshigoda, "Implementing scrum in a distributed software development organization," in *Agile Conference (AGILE), 2007*, Aug 2007, pp. 371–375.
- [12] "Jira agile," www.atlassian.com/software/jira/.
- [13] "Ibm - rational team concert," www.ibm.com/software/products/en/rtc/.
- [14] "D3.js - data driven documents," <http://d3js.org/>.