

# DRIVING SECURE COLLABORATION AND E2E TRANSPARENCY WITH IMPLEMENTATION OF TOOLS IN LARGE SCALE AGILE SOFTWARE DEVELOPMENT LIFE CYCLE

**Gopalakrishnan Sriraman**

*Sona College of Technology*

*Junction Main Rd, Suramangalam, Salem, Tamil Nadu 636005*

**Dr. J. Senthil Kumar**

*Professor*

*Department of Information Technology*

*Sona College of Technology*

*Anna University*

## **Abstract**

*Agile software development is becoming the defacto software engineering approach in the modern world where the requirements are highly dynamic and software development process has been highly globalized. One of the key success factors of global agile software development is the collaboration effective collaboration of teams. The original agile teams were mostly co-located with business and the concept and practices were more suited for the collocated, self-contained development teams. However as Agile matured and the need to leverage low cost and best skilled resources across globe collocation of resources is no longer possible. This leads to a critical challenge on collaboration; the business, developers, testers and end users are highly distributed no longer can rely on face-to-face discussions, drive agile ceremonies, and do efficient agile development. Each stakeholder in the process may have different collaboration needs and to address this challenge we will need to leverage and extend available digital tooling in the market place. This paper deals with analysis of available tooling and implementation of a suite of Microsoft applications to meet the needs of collaboration and end-to-end transparency in large-scale agile software development programs. The major requirements considered are driving end-to-end agile processes from requirement planning sessions, daily management ceremonies to demo and user acceptance and end-to-end traceability of software to requirements and test evidences. Microsoft Teams together with Azure DevOps suite components including Azure Boards, Pipelines, and Test Plans were implemented in combination to address the needs. White spot analysis is done on this platform and the gaps were addressed with extending the solution with custom extensions. This secure collaboration and project management platform addresses the business need in bridging the collaboration gap and bring the must-needed transparency across the delivery life cycle, thereby improving the business outcome and efficiency.*

*The decentralized code repository management implemented through Azure Repos and its integration with Boards and Test plans help bring the much required transparency*

*from requirements to build components to Test scripts , defects and deployment pipelines.*

*This solution is also future proof and cost effective as this complete solution is based on PaaS and SaaS components and complete infrastructure is built in cloud*

**KEYWORDS:** *Agile Tools: Collaboration: Large Scale Agile: DevOps:*

## 1. Introduction

Introduction Agile methods are undeniably one of the most important and recent developments in software engineering. Over the past 30 years they have highly boost success rates in software development, motivation and productivity of IT teams and also improve agility to market and quality. Spontaneous market changes have changed the business plans for most companies, so they had to shorten the delivery of a product and to be more responsive to the expectations and demands of customers. Agile development methodologies are very useful tools, representing a real support to the development of software processes, being able to adapt according to the new market trends. Implementing agile methodologies have a huge impact on the discipline of business analysis in general, since they put in a forefront a new method of approach and prioritization of requirements within a team, which is implementing an IT project. Successful management of an agile project depends heavily on the process of defining requirements, developing smaller steps which means a closer collaboration with the whole team along SDLC (Software Development Life Cycle) of the project. Agile is about innovation and nowadays most companies operate in highly dynamic environments, which mean they need tools that can adapt quickly to changes. They do not need just new products and services but also innovation in their business processes.

Agile is mainly a set of values and principles, and unless one can imagine ways of organizing a project team around these values and principles we can say that person is not agile. This is very important to understand whereas many are rigid in supporting a unique way to be agile. In one sense, agile methods are an outgrowth of the plentitude of rapid development methodologies emerging from the 1980s. From my point of view, each has its own vision of what agile means and often it is hard to share this strategy with others, showing a high resistance to change when the direction of agility is not anticipated. Since the research is about agile project management tools, In this section I will focus the discussion on tools that would provide a full capability for handling large and complex enterprise projects. Tool Features Agile project management tools are fundamentally different from traditional ones, because we talk about dynamic and adaptive approach, based on the concept of a continuous management flow of the project. As agile projects become larger and more complex, it becomes essential to use specialized tools to manage and share information quickly and efficiently. There are a lot of commercial and open source management tools, ranging from sophisticated web-based applications to simple and specialized utilities.

To keep a better comparison the following features maximizing business value were analyzed: Agile reporting and metrics – Collaboration- Quality assurance - Easy-to-understand progress reports for stakeholders - Integration with the company's existing system, Communication - Communicate updates within teams - Share task lists, assignments and feedback o Project assessment - Identify and remedy project incidents - Evaluate constantly performance - Estimate financials

Several researches have examined the needs for agile tooling for effective delivery using digital and physical tools. The Physical tools though brings face-to-face interactions, are not suitable for large projects and programs wherein the resources are highly distributed.

In addition, the advancement in IT like cloud, artificial intelligence and automation has accelerated the way software is delivered. DevOps approaches like Continuous Integration, Continuous Testing and Continuous deployment help accelerate the software development process and improves overall cycle time and quality.

## 2. Comparison of various Tools from Market

Various applications are analyzed to fit the business requirements and four applications namely Azure DevOps, Rally TFS and IBM RTC were short-listed.

### 2.1. Criteria considered

The parameters considered are grouped as

- Functional parameters – The ability to support Project life cycle processes and methods
- Technical Parameters – Underlying IT components and architecture
- Non Functional Requirements (NFRs) – The availability, performance, security and compliance of the system
- Future proof – ability of the application to scale up for future business needs
- Operational Parameters – cost of building and maintaining the application.

### 2.2. Comparison results across tooling

A detailed study and comparison was done on these four applications based on Technical, business, operational parameters as below.

Proof Point	Comment	Azure DevOps	Rally	TFS	IBM RTC
Feature release plan	The Release Planning Board app provides program managers with a view of portfolio items ready to be scheduled to a release. Use the board to prepare for and carry out mid-range planning. Release Planning facilitates release preparation and the planning process at the program level.	3	3	2	3
Feature		3	3	2	3

mutations (add-edit-delete)					
Portfolio items	The portfolio item hierarchy models your portfolio where each level represents a different type of strategic goal. The higher types of the hierarchy represent goals to bring to market over a longer period of time. The lower types of the hierarchy represent more discrete items of market value.	4	4	2	4
Portfolio item timelines	The Portfolio Item Timeline app allows you to visualize how portfolio items are progressing over varying time intervals.	4	4	2	2
Epic mutations (add-edit-delete)	The Epic Progress app displays a high-level view of the children associated with epic stories in a selected release. Use this app to help you determine whether:	4	4	2	3
Portfolio items view & board	The Portfolio Item Timeline app allows you to visualize how portfolio items are progressing over varying time intervals	3	3	2	2
Portfolio hierarchy	Use the Portfolio Hierarchy app to visualize and manage the relationships between different levels of portfolio items and the stories that support them. This is a useful app for project managers and product owners.	4	4	2	3

Capacity planning	built for the program and portfolio level of enterprise scale Agile. With the new Capacity Planning page in the platform, strategic planners can rapidly translate their business initiatives into realistic and adaptable action plans, without relying on disconnected, manually updated spreadsheets.	4	4	3	4
Portfolio Kanban	The Kanban Board app provides teams with the option of managing pull-based, lean software development projects.	5	4	4	4
Customized Kanban	The Custom Board app provides you with a way to view and edit work items.	5	3	4	3
Backlog view	Views are a collection of like data that are arranged in a relevant display. Some views are standard static views while other views can be customized	3	3	2	3
Iteration Planning board	provides your team with a view of the user stories and defects ready to be scheduled, and any work items scheduled in the next three iterations.	4	4	3	2
User stories view	View the User Story summary and detail pages	4	4	3	2
User stories mutations (add-edit-delete)	Editable Detail Pages (EDPs) that allow you to directly edit fields. If you can see a field and have the right permissions, you can edit it. Fields are automatically saved when you leave a field.	3	3	3	2
Iteration status Kanban	Iteration Status page provides your team with a view of work items within	4	4	2	4

	an iteration organized by schedule state.				
Iteration Scope Change	The Iteration Scope Change app displays all work items (user stories, defects, and defect suites) that have been added or removed from an iteration, giving a succinct view of whether scope is changing during the course of an iteration.	4	4	3	2
Task board	The Task Board app provides you with a way to view and update tasks during your daily standup meeting.	5	4	4	4
Tasks	Manage and view your tasks Create a task Edit a task Copy a task Copy tasks from another work item Assign a task Import tasks Use task fields	5	4	3	3
Task board	The Task Board app provides you with a way to view and update tasks during your daily standup meeting.	5	3	2	3
Manage Test cases	Created from a work item which creates an automatic association between the test case and the work item (recommended). Created as a stand-alone test case. Edited to add information or update changes as your requirements evolve.	4	4	3	4
Manage Test plan	The Test Plan summary page provides testers with a single location to organize and manage test cases into test folders. When planning your test effort, use test plans to	3	3	2	3

	assemble test cases in a hierarchy of folders, run the tests in groups, log test results and defects, then see the summarized results.				
Test case addition	Created from a work item, which creates an automatic association between the test case, and the work item (recommended). Created as a stand-alone test case. Edited to add information or update changes as your requirements evolve.	3	3	2	3
Manage Defects	Managing defects includes the following:  Use the Defect summary and detail pages Create a defect Edit a defect Manage duplicate defects Convert a defect to a user story Schedule a defect Delete a defect My Defects Defect fields	3	3	2	2
Release defect trend	The Defect Trend app allows you to view cumulative defects opened versus closed over time, showing whether the trend is increasing or decreasing. The chart shows the rate at which defects are being activated (red line) and in a state other than closed, the rate at which defects are being terminated or closed (green line), and the trend for the total number of active defects (black line).	4	4	2	2
Release planning	To plan a release, use the Plan page to organize your backlog and release backlog. Work items in the Plan and Release Backlog sections pertain only to the	5	3	3	3

	currently-selected project.				
Release tracking	Release metrics Release defect status Release task status Release test case status	3	3	2	2
Release planning Kanban view	To plan a release, use the Plan page to organize your backlog and release backlog. Work items in the Plan and Release Backlog sections pertain only to the currently selected project.	5	3	4	3
Release Scope Change	The Release Scope Change app displays all work (user stories, defects, and defect suites) that have been added or removed from a release, giving a succinct view of whether scope is changing during the course of a release.	5	3	3	3
Release Planning board	The Release Planning Board app provides program managers with a view of portfolio items ready to be scheduled to a release. Use the board to prepare for and carry out mid-range planning. Release Planning facilitates release preparation and the planning process at the program level.	4	4	2	3
Track iteration status	Iteration Status page provides your team with a view of work items within an iteration organized by schedule state.  Features of the Iteration Tracking app includes:  Easily switch between grid and board views to view and manage iteration details.  Visualize iteration data in multiple ways using charts.	4	4	2	4



	Make rapid changes within a card or grid line without opening the detail page. Personalize which columns of data to display in the grid view.				
Team status	The Team Status page displays all project team members in bold font, with their assigned tasks and individual capacity by project for the selected iteration. When using a project hierarchy, this page reflects any view scoping settings that you have indicated in the project picker.	5	3	2	3
Release status	Release metrics Release defect status Release task status Release test case status	4	4	3	3
Release metrics	The Release Metrics page summarizes work items that directly impact your progress toward achieving your milestones. Use this information to help you plan possible future resource needs, identify trends, determine software quality, and help predict velocity.	3	3	2	2
Custom reports	Reporting capabilities to track your team's progress with standard reports on the Reports tab. You can also create custom reports based on your development data to make fact-based, informed decisions. There is no limit to the number of custom reports you can create.	4	4	2	3
Insights	Ability to compare your own performance against a benchmark of over 13,000 teams. Compare the performance of a single team (or program, or entire	3	3	3	2

	organization) to its history ... , compare it to the average of teams in the workspace ... , or compare it to an industry benchmark.				
Custom dashboards	Your dashboard shows you – at a glance – the information that is most important to you across projects, teams, and workspaces.	4	4	2	2
Iteration summary	The Iteration Summary app allows you to see a simple overview of team progress against an iteration. Based on the status of work, defects, and test cases, you will be able to see colored indicators that help address problems as they arise.	4	4	2	4
Release burnup charts	The Release Burnup app displays work delivered so far in the release to proactively anticipate whether the release scope will be delivered.	4	4	3	4
Iteration burndown	The Iteration Burndown app displays work remaining and completed in the iteration to proactively anticipate whether the committed work will be delivered by the iteration end date. It is also useful during iteration retrospective meetings, to help identify events during the iteration or problems with estimation during planning.	5	3	2	3
Custom velocity tracking app (determine avg velocity)	The Velocity Chart app displays trends in value delivery, including work accepted after the iteration and completed work not yet accepted.	4	4	3	2

Milestone burnup	The milestone burnup chart shows how work is tracking to the target date. The blue line indicates scope, and the green bars are the accepted work on a given date.	4	4	3	3
Discussion fields	<p>communicate and collaborate on shared work. Depending on your working agreements, your team may choose to collaborate using fields inside of work items, apps, shared page views, shared pages and apps, or a combination of these options. Below are some recommendations on how to utilize each feature.</p> <p>Collaborate with team members includes the following:</p> <p>Discussions Attachments Dashboards Custom views Shared pages and apps</p>	4	4	3	3
Revision tracking	Each time a change is made, the revision number of the work item is automatically incremented, and the date, time, and the name of the team member who authored the change are inserted into the history. Use the Change Description field to add a comment or description of the change in the editor window. Both the system-generated and user comments display on the Revisions page.	5	3	3	3
Custom app	build your own custom apps to suit your needs, including:	5	4	3	4

	Custom Grid Custom HTML Custom URL				
Custom dashboard	The Custom List app (formerly known as the Custom Grid) is a powerful tool for helping manage work across enterprise teams. These pages help focus in on and manage the work most cared about, while also communicating real-time status with the rest of the organization. You can use the Custom List app to prepare, plan and track portfolios, programs and development work.	4	4	3	3
<b>Architecture</b>					
Technology fit	Will either of the tools be impacted by target architecture?	3	3	2	3
Integration capabilities	What are the integration capabilities of the solution?	5	4	2	4
<b>Delivery / Operations</b>					
Ease of Use	How user intuitive is your tool? What training needs are required to start using your tool? What training is available?	4	3	2	4
Ease of transferring historical data	Explain how data can be transferred from adhoc tools to your tool in an automated fashion with preserving necessary relationships (Theme-Epic-User Story-Tasks / Team / Release)	4	5	2	3
Reporting capabilities & link to Insight Center	What OTB reporting capabilities does the tool have? What OTB data export capabilities are available? What customization is possible to create customized reports?	4	3	2	3

	Link to Insight Center?				
User management & authorizations	Explain what user management & user authorization capabilities does your tool have	3	3	3	3
Change management to transition to new tool - timeframe / impact on running operations	What is your expectation wrt timeframe / effort / impact required to transfer from adhoc tools to your tool?	4	2	4	4
Serviceability	What is required to support the solution after its implementation?	5	3	2	4
Availability	What is the availability of the tool? What regular maintenance is planned? What downtime (if any) is required for regular maintenance?	5	3	4	4
SLAs	What SLAs are available for Sev 1, Sev 2, Sev 3, Sev 4 issues raised?	4	3	4	4
<b>TCO</b>					
What is the impact on license-/maintenance costs		5	2	4	4
What is the impact on the hosting costs		5	2	3	3
What is the impact on implementation costs		4	5	4	4
What is the impact on operations costs		5	2	2	4
<b>Total score of solution</b>		<b>241</b>	<b>204</b>	<b>156</b>	<b>183</b>

### 2.3 Implementation and Proposed Architecture

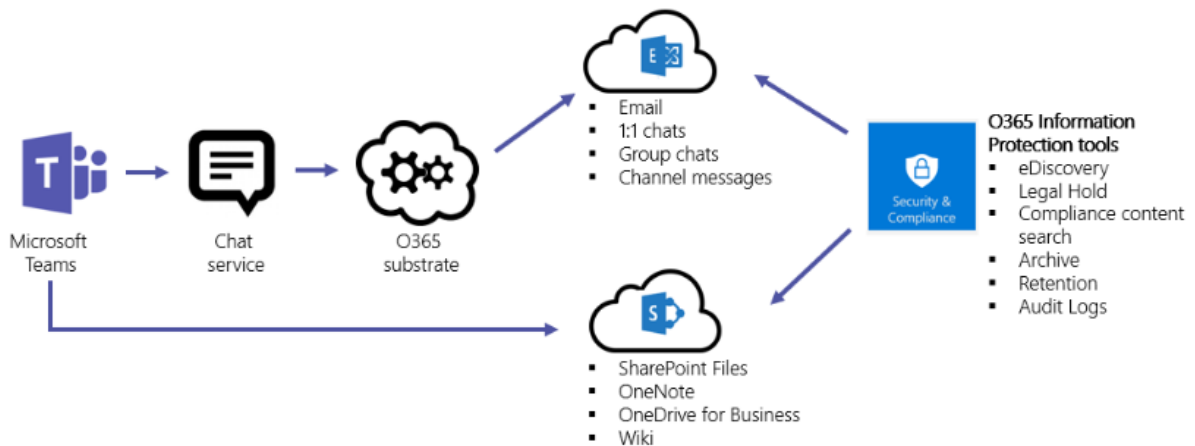
Implementing Azure DevOps as the foundation of the Agile tooling and collaboration platform, the white spots that were required for specific process needs were addressed by building custom components and also integration various tools that brings in the capabilities.

Below is the reference architecture proposed and implemented to cover the complete gamut of business needs in addressing the objective.

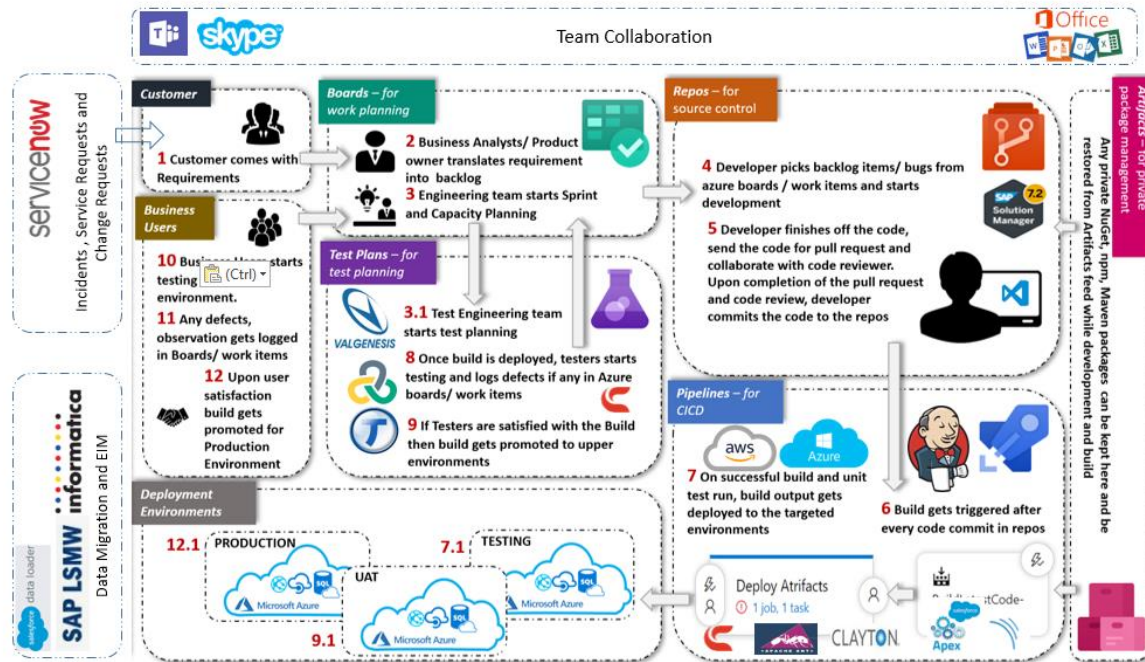
Microsoft Teams is chosen as the Collaborative platform to address the close net collaboration and information sharing needs as it is closely integrated with Azure DevOps.

Microsoft Teams is a chat-based platform with inbuilt connectors that can be extended to connect with various other applications and office platform.

The data in Microsoft teams is secured with enterprise level data privacy architecture and policies.



The DevOps components that are used for continuous intergation and Continuous deployment are tied to MS Teams and Azure DevOps to bring an end to end traceability and transparency. An reference architecure is shown below



### Conclusions

Agile and collaboration tools implementation in Large-scale Agile programs improved the rate of success of the projects and bring in the required collaboration and efficiency.

The Agile platform also brings in a highly compliant and E2E transparency of requirement-Build-deployments and helps improve serviceability and maintainability of the products.

The implementation helped bring business, IT and customers to collaborate in one platform that in turn builds confidence and improves NPS/Delivery satisfaction for stakeholders. With building Continuous Integration and Continuous deployment capabilities, the overall time taken from requirement gathering to deployment can be cut down by 50% with automating the processes and deployment practices.

### Acknowledgments

I acknowledge the support from Microsoft Azure Product Management team in providing the support and also clarifications, guidance across the duration of this implementation.

## References

1. *12th Annual State of Agile Survey Report, VersionOne, 2018*; <http://stateofagile.versionone.com>.
2. <http://agilemanifesto.org/> - *The guiding manifesto and principles of Agile software delivery*
3. <https://martinfowler.com/articles/newMethodology.html> *An alternative way of delivering software*
4. <https://www.ibm.com/in-en/marketplace/change-and-configuration-management>
5. <https://visualstudio.microsoft.com/tfs/> - *Microsoft Team foundation server documentation*
6. <https://azure.microsoft.com/en-in/blog/introducing-azure-devops/> *Microsoft Azure DevOps documentation*
7. <https://www.ca.com/us/products/ca-agile-central.html> *CA Agile central documentation*
8. *Survey: "The State of Agile Development," VersionOne, 2007*; <http://stateofagile.versionone.com>.
9. M. Aoyama, "Web-Based Agile Software Development," *IEEE Software*, vol. 15, no. 6, 1998, pp. 56–65.