Table 8. Are we more productive working agile?

Name	Description	Calculation Formula
Velocity		Quantity of story points completed for work items of a type over a period of time
Blocked Tasks		Number of tasks blocked during a specific time period
Function Points per Man-Year		Amount of work done by an individual throughout the year
Burndown Chart	Measures sprint progress and provides indicators of the team's work process	Marks the sprint days on the horizontal axis and the points planned to compose the sprint on the vertical axis, starting from the maximum points of the sprint (team velocity) to zero
Burnup Chart	Measures progress based on remaining hours or points from the top down. Measures release progress and provide indicators of the team's work process	Marks the sprint days on the horizontal axis and the points planned to compose the sprint on the vertical axis, starting from the maximum points of the sprint (team velocity) to zero
Velocity Deviation	Measures velocity stability	$\sigma(V)/\mu(V)$. Where $\mu(V)$ is the expected velocity (average velocity based on a certain number of previous sprints) and $\sigma(V)$ is the velocity standard deviation
Dependency Count	Dependency between teams where bugs and delays affect immediate dependents	Number of immediate dependents
Efficiency	Indicates how well we use the team compared to maximum capacity	*AH / C*. Where *AH* represents the actual hours spent producing tangible results and *C* is the ideal number of hours a team can deliver, depending on team size, number of teams, non-working days and days spent on ceremonies. $C = S*(DS-(NWD+(DS*KO/20)))*TS*DH)$, where *DS* is the sprint duration, *NWD* represents days off, *KO* stands for knowledge transfer/planning/estimation days, *TS* is the team size and *DH* is daily development capacity in hours
Impediments	Any 'time loss' due to a defect or obstacle that hampers productivity	Number of hours that do not produce tangible results
Commit Pulse	Measures how continuous integration is within sprints	Number of days between commits
Planned Velocity		Amount of work (story points) that a team expects to complete during a sprint
Cost per function		*Task cost / Number of task functions*
Remaining Cost	Measures the remaining cost for task completion	*Task cost - Expense cost*
Cost Spent (All Tasks)	Percentage of cost spent during task implementation	*(Expenditure cost / Task cost) * 100*
Remaining Functions		*Task functions - Task functions completed*
Completed functions (All Tasks)	Percentage of functions executed during task implementation	*(Completed functions of the task / Total functions of the task) * 100*
Remaining Patterns	Measures the remaining patterns for the task	*Task patterns - Patterns achieved*
Achieved patterns (All Tasks)	Percentage of patterns achieved during task implementation	*(Achieved patterns / Task patterns) * 100*
Available Cost	Measures the available cost for subtasks or main tasks	*Task cost - Expense cost*
Deployment Frequency	Measures the productivity and flow speed for an organization's DevOps resources	Number of deployments in a given time period
Acceleration	A productivity metric that will also reflect how efficiently the team adapts to changes	*(Team velocity in iteration X - Team velocity in iteration Y) / Team velocity in iteration Y*