

Agile Project Management Metrics



Agile Spirit Meetup
25 September 2017

Rachel BenHanoch

“

If you cannot measure it,
you cannot improve it

(Attributed to Peter [Drucker](#) or Edwards [Deming](#))

“

**If you don't know where you
are heading,
you will not get there**

Rachel BenHanoach



**Data is as strong as your
ability to act on it**

Rachel BenHanoach

What's in it for me



The Basic Metrics

The Advanced Metrics

The Extra-Mile Metrics

The Don't-Use Metrics

Hi, I am Rachel

- **B.Sc Computer Science + MBA : Hebrew University**
- **CSM and CSP Agile Alliance certifications**
- **20 years in the high tech industry**
- **Served in executive roles in the areas of Product management, Project management & Development**
- **Worked with diverse types of products in many Startups and Enterprises**
- **Lead and managed the transition to Scrum in the last 4 companies I worked for**
- **About 110% of my time I am Guesty's VP Product**
- **At the rest of the time I am an Agile advocate, a lecturer in Shenkar, and a mother to 4 great kids**

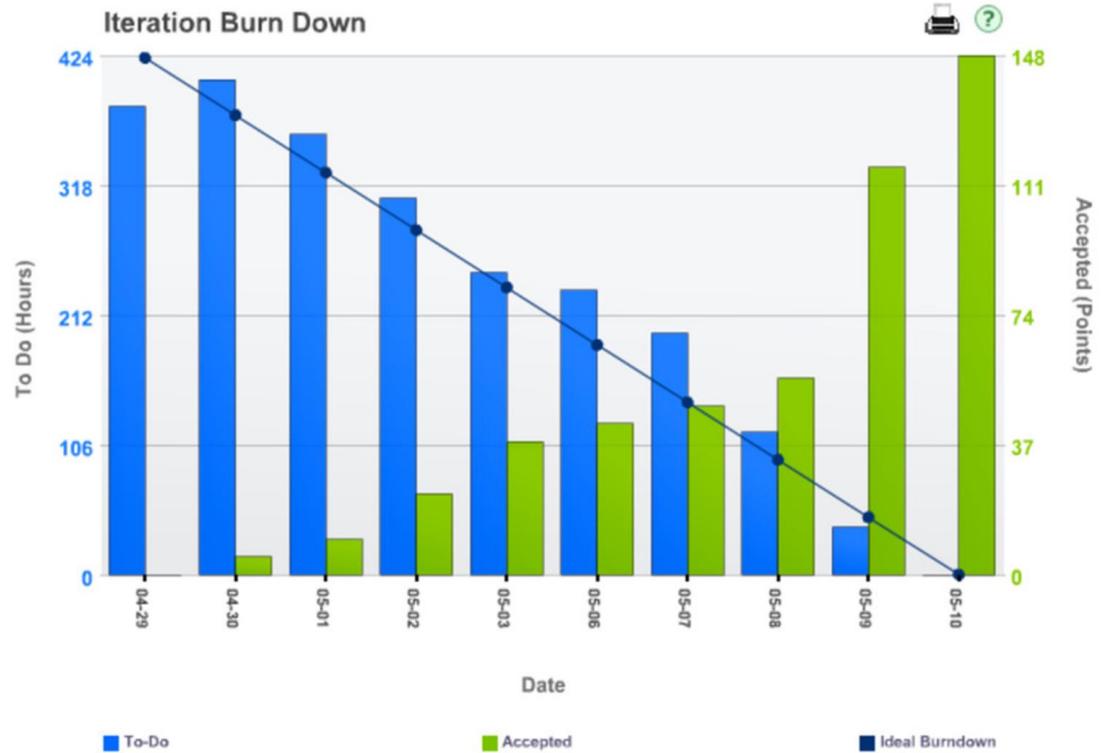
Fast Back
KOMELON 100 ft.



The Basic Metrics

Burndown chart

Burndown chart shows the trend of story points left to complete



Velocity

Velocity is a measure of the amount of work a Team can tackle during a single Sprint.

Velocity is calculated at the end of the Sprint by totaling the Points for all fully completed User Stories.

++ Velocity Variance

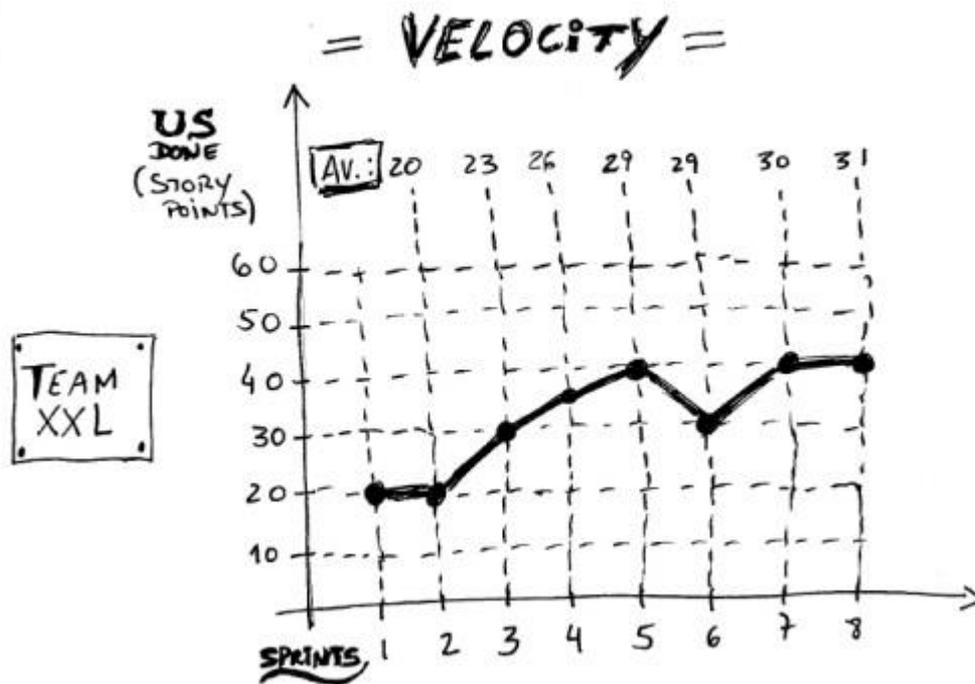


Image source: [Catia Oliveira](#)

Include incomplete user stories?

No

Include bugs?

Yes

Should I compare the velocity of different teams?

No



The Advanced Metrics

The 4 metrics for Planning / Estimation → Predictability

Completed Story Points

Measure planned vs Actual of completed story points.

It is also valuable to measure how many S/M/L items made it into the sprint successfully.



The 4 metrics for Planning / Estimation → Predictability

Completed Story Points

Measure planned vs Actual of completed story points.

It is also valuable to measure how many S/M/L items made it into the sprint successfully.

Sizing Accuracy

Expected size of story versus actual size - planned vs actual. This is usually discussed in the Retrospective and not kept in records. If tracked, measure the average of the distance of the actual size from the original estimate.



The 4 metrics for Planning / Estimation → Predictability

Completed Story Points

Measure planned vs Actual of completed story points. It is also valuable to measure how many S/M/L items made it into the sprint successfully.

Sizing Accuracy

Expected size of story versus actual size - planned vs actual. This is usually discussed in the Retrospective and not kept in records. If tracked, measure the average of the distance of the actual size from the original estimate.

Task Type Distribution

% of work devoted to features, bugs, technical-debt, infrastructure, support, and unplanned emergencies.



The 4 metrics for Planning / Estimation → Predictability

Completed Story Points

Measure planned vs Actual of completed story points. It is also valuable to measure how many S/M/L items made it into the sprint successfully.

Sizing Accuracy

Expected size of story versus actual size - planned vs actual. This is usually discussed in the Retrospective and not kept in records. If tracked, measure the average of the distance of the actual size from the original estimate.

Task Type Distribution

% of work devoted to features, bugs, technical-debt, infrastructure, support, and unplanned emergencies.

Effective Hours

How many effective hours a day do we really have. Useful when using sizing in hours.



The 5 metrics for Quality

Bugs Trend

#bugs_closed vs #bugs_opened
- not taking into account bugs
that were opened and closed
during the sprint



The 5 metrics for Quality

Bugs Trend

#bugs_closed vs #bugs_opened
- not taking into account bugs that were opened and closed during the sprint

Sprint Quality / Quality Debt

Newly introduced bugs.
Total number of newly created bugs that were found in the sprint by the internal team, and are carried forward. Note the severity.



The 5 metrics for Quality

Bugs Trend

#bugs_closed vs #bugs_opened
- not taking into account bugs that were opened and closed during the sprint

Technical Debt

Quantify in story points the value of work not done, in the interest of pushing deliveries.

Sprint Quality / Quality Debt

Newly introduced bugs.
Total number of newly created bugs that were found in the sprint by the internal team, and are carried forward. Note the severity.



The 5 metrics for Quality

Bugs Trend

#bugs_closed vs #bugs_opened
- not taking into account bugs that were opened and closed during the sprint

Technical Debt

Quantify in story points the value of work not done, in the interest of pushing deliveries.

Sprint Quality / Quality Debt

Newly introduced bugs.
Total number of newly created bugs that were found in the sprint by the internal team, and are carried forward. Note the severity.

Product Approval Index

How many stories were approved by product As-Is - without the need to release with less content or lower quality. The index is the percentage of stories approved with flying colors.



The 5 metrics for Quality

Bugs Trend

#bugs_closed vs #bugs_opened
- not taking into account bugs that were opened and closed during the sprint

Technical Debt

Quantify in story points the value of work not done, in the interest of pushing deliveries.

Automation

Percentage of test cases automated

Sprint Quality / Quality Debt

Newly introduced bugs.
Total number of newly created bugs that were found in the sprint by the internal team, and are carried forward. Note the severity.

Product Approval Index

How many stories were approved by product As-Is - without the need to release with less content or lower quality. The index is the percentage of stories approved with flying colors.



The 4 metrics for Risk Reduction

Shared Resources Risk mitigation

Measure expected load on shared resource and ensure they are not spread too thin (Examples: Design, DevOps)



The 4 metrics for Risk Reduction

Shared Resources Risk mitigation

Measure expected load on shared resource and ensure they are not spread too thin (Examples: Design, DevOps)

Work Item Status

Check if items are not moving smoothly between stages. Closer to the end of the sprint use resources to push to 'Done' items that are likely to get done rather than start new items.



The 4 metrics for Risk Reduction

Shared Resources Risk mitigation

Measure expected load on shared resource and ensure they are not spread too thin (Examples: Design, DevOps)

Early Defect Detection

How many bugs are open each day of the sprint. Follow trend to ensure bugs are found early rather than later. It indicates both if stories are ready for QA early enough in the sprint as well as the trend of the quality within a sprint.

Work Item Status

Check if items are not moving smoothly between stages. Closer to the end of the sprint use resources to push to 'Done' items that are likely to get done rather than start new items.



The 4 metrics for Risk Reduction

Shared Resources Risk mitigation

Measure expected load on shared resource and ensure they are not spread too thin (Examples: Design, DevOps)

Early Defect Detection

How many bugs are open each day of the sprint. Follow trend to ensure bugs are found early rather than later. It indicates both if stories are ready for QA early enough in the sprint as well as the trend of the quality within a sprint.

Work Item Status

Check if items are not moving smoothly between stages. Closer to the end of the sprint use resources to push to 'Done' items that are likely to get done rather than start new items.

Happiness Index

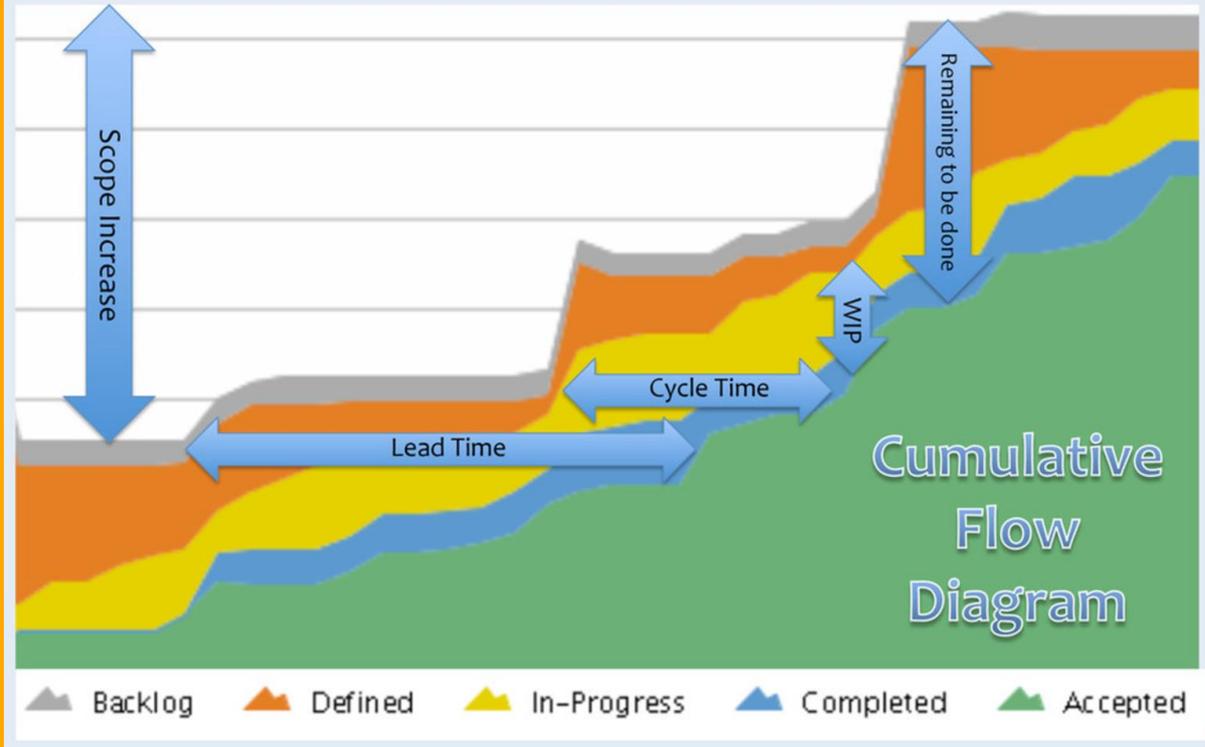
Ask team members to indicate "Yes" or "No" on a sticky note, or use a scale of 1-5. Ensure to open a discussion to understand what keeps the team happy less than 100%

Source: [Venkata Kalluri](#)



Productivity

How is our throughput?





The Extra Mile Metrics

Quality/Defect containment

Defect containment (DC) = $[1 - (Dc/Di)] * 100\%$

Where:

Dc = Number of P1 and P2 defects detected by customers for a release

Di = Total number of P1 and P2 defects detected internally

Keep track and count the number of defects reported by the customer per release. This can be a negative value depending on the number of defects reported by the customer. Care must be taken to include only the defects that are relevant to the product.

Feature waste

Feature waste (FW) = $[1 - Fu/Fa] * 100\%$

Where:

Fu = Number of features, or effort spent on features, used by customers in a release

Fa = Total number of features or effort spent on features delivered in the release

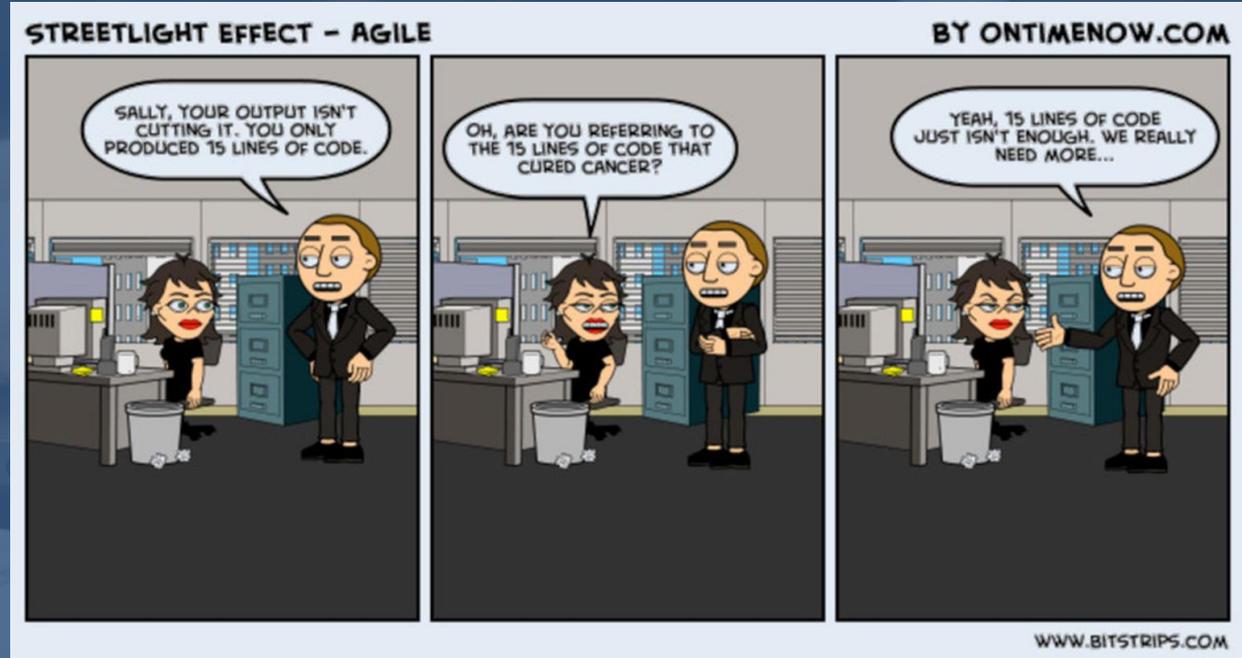
A measure of the amount of effort spent by the feature teams on features that are "wasted" (features that the customers are not interested in). This is an important metric, as elimination of waste is one of the big ideas in Agile





The Don't-Use Metrics

KLOC (Kilo Lines of Code)



Professional agilists use Agile Metrics

For the Scrum Master

- Agile metrics provide the "pulse" of the team's progress and productivity
- Change is better embraced, given the insight provided by metrics

For the Product Owner

- With Agile metrics, the product owner has solid and detailed information to relay to the stakeholders

For the Team

- When metrics are known, the development team can better assess where to direct its attention and which tasks to prioritize

Back to basics

At the Sprint level:
Working software is the primary measure of success

At the Release level:
Satisfied customer is the primary measure of success

WORKING
SOFTWARE



How likely are you to recommend
Guesty to a friend?

 0 1 2 3 4 5 6 7 8 9 10

Not likely

Very likely

Take Away

If you cannot measure it, you cannot improve it

If you don't know where you are heading, you will not get there

Data is as strong as your ability to act on it

Implement the Essential measures, and 3 more of the Advanced ones.

Measure the start point and set Targets.

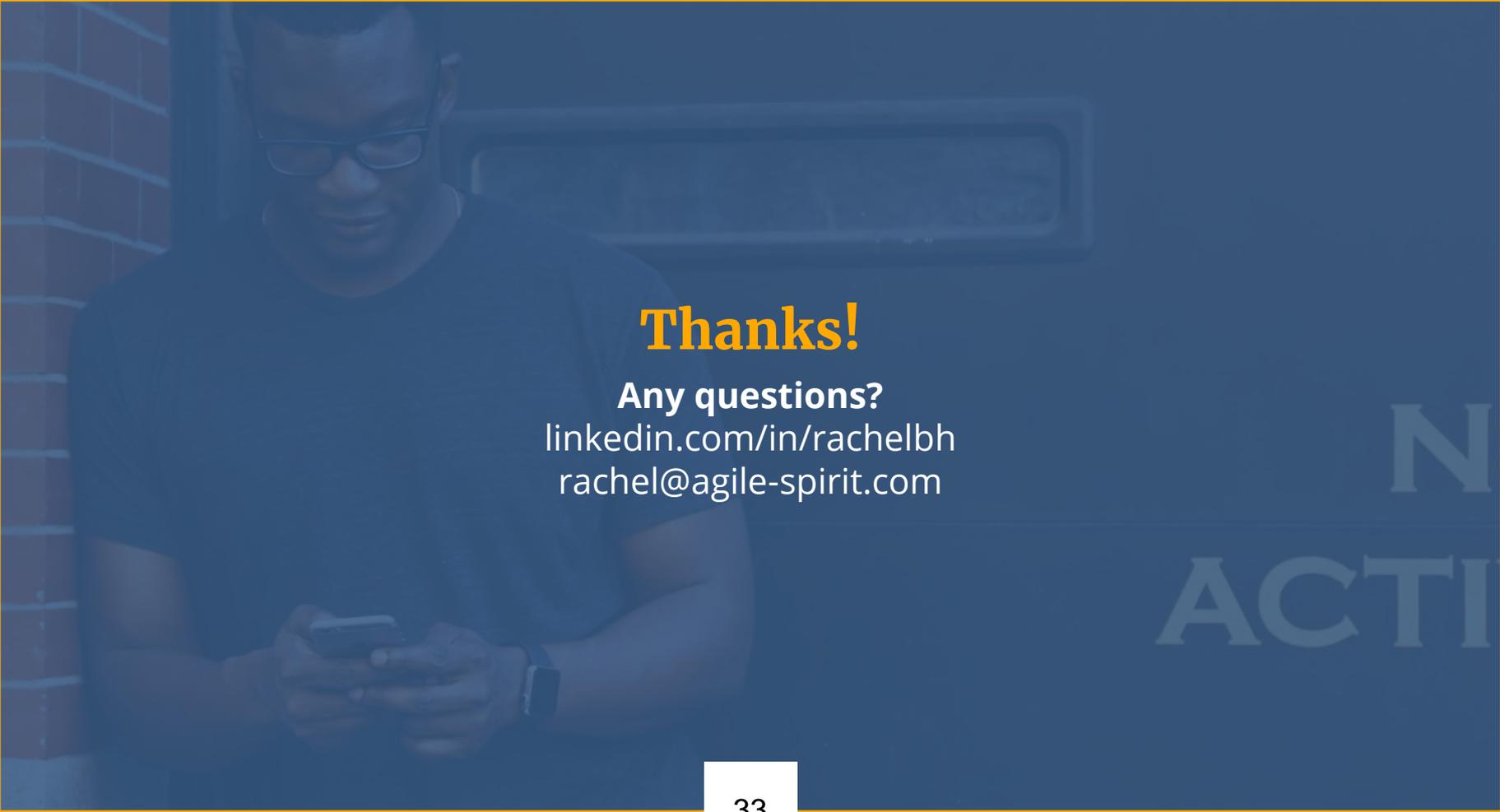
When you've reached your goals, set new targets.

Track the trends, review in retrospectives, and act upon it to improve continuously.

Credits & Resources

- [The Significance of Agile Metrics](#), Scrum Alliance, 2014
- [Agile Metrics for Organizational Productivity](#), Scrum Alliance, 2014
- [Agile Metrics and Business Value](#), Scrum Alliance, 2015
- [Key Metrics for Agile Teams](#), Scrum Alliance, 2017
- [Important Metrics for Measuring Scrum Processes](#), Scrum Alliance, 2017
- [How to Calculate and Use Velocity to Help Your Team and Your Projects](#), Catia Oliveira, 2014

- *Presentation template by [SlidesCarnival](#)*
- *Photographs by [Startup Stock Photos](#)*

A man with glasses is looking at his smartphone. The background is a blue-tinted image of a brick wall and a door. The text is overlaid on the right side of the image.

Thanks!

Any questions?

[linkedin.com/in/rachelbh](https://www.linkedin.com/in/rachelbh)

rachel@agile-spirit.com