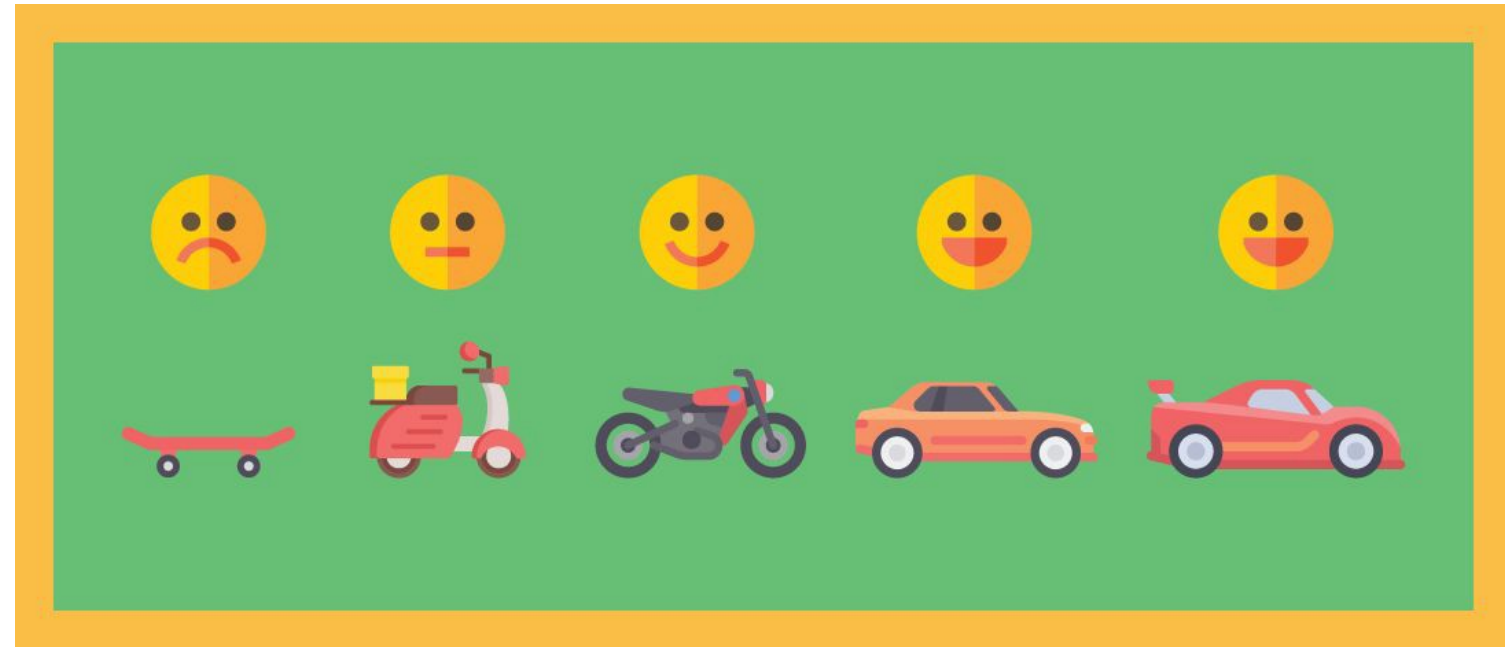


# Project Methodology

Agile methodology underpins not only our project management approach, but our project mindset

# We use Agile. What does that mean?



Source: dzone.com

1. Start with broad-stroke requirements (Usually in the form of user stories)
2. Build a Minimum Viable Product (MVP)
3. Review and collect feedback
4. Implement changes based on feedback
5. Repeat

Traditional Project Management	Agile Project Management
All features and requirements are fixed at the start of the project.	Project starts with a backlog of desired features. Items are selected on a sprint-by-sprint basis from the backlog.
Specifications are planned in detail before starting work.	General functionality is defined for each backlog task, but specifics are flexible.
Adding or changing features after the project has started requires a formal change order.	Adding or changing features can be done any time, by managing the backlog.
Timeline is fixed.	Timeline is flexible.

# User-Driven Agile

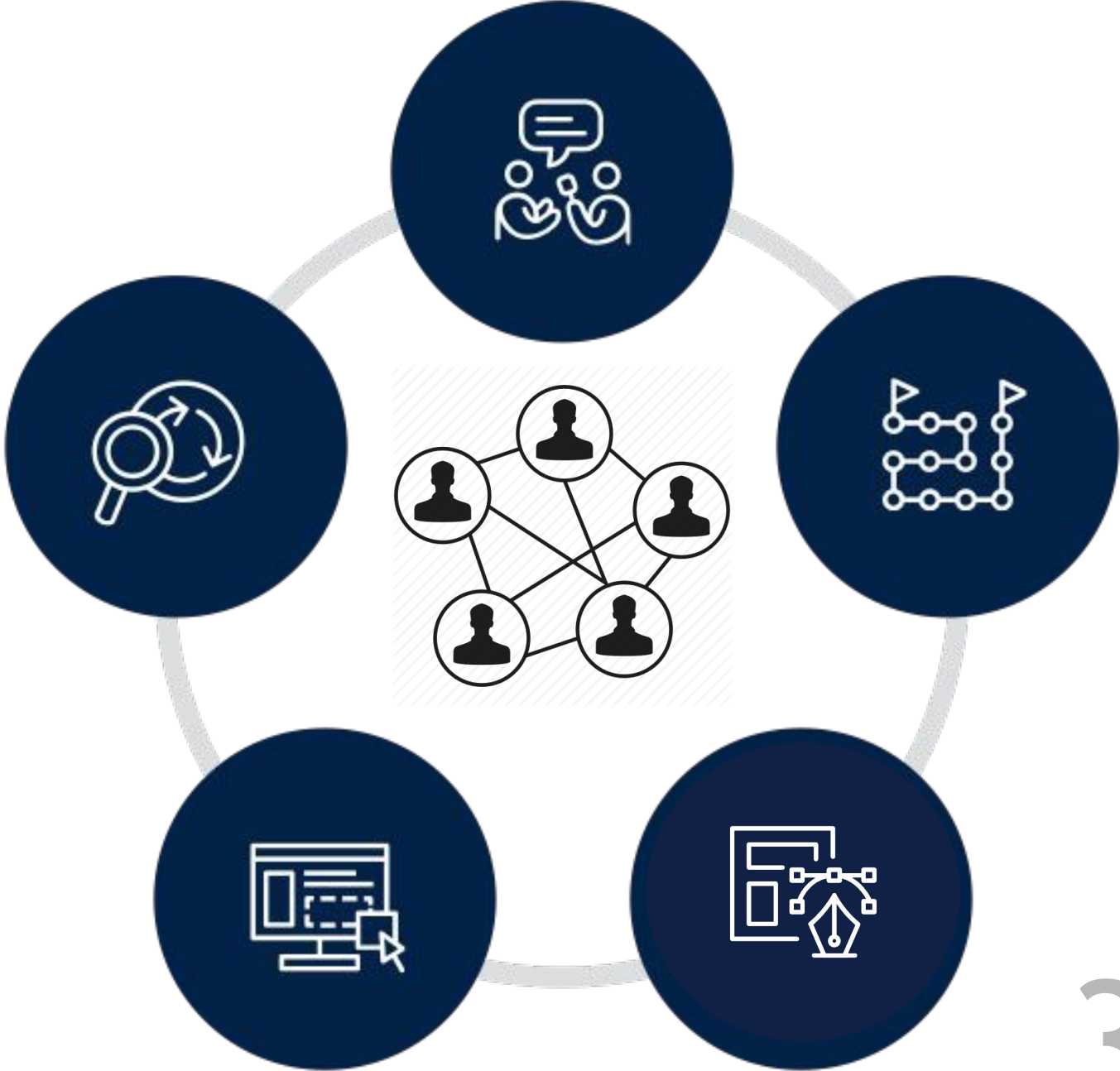
**1 Discover**  
Knowledge transfer and research. Understanding both business goals and user goals.

**2 Plan**  
Reviewing and analyzing our findings and planning out the overall implementation strategy.

**3 Design**  
Realizing the strategy and concept and turning it into a tangible design and prototype.

**4 Build**  
Implementing and deploying the project based on the strategy and design.

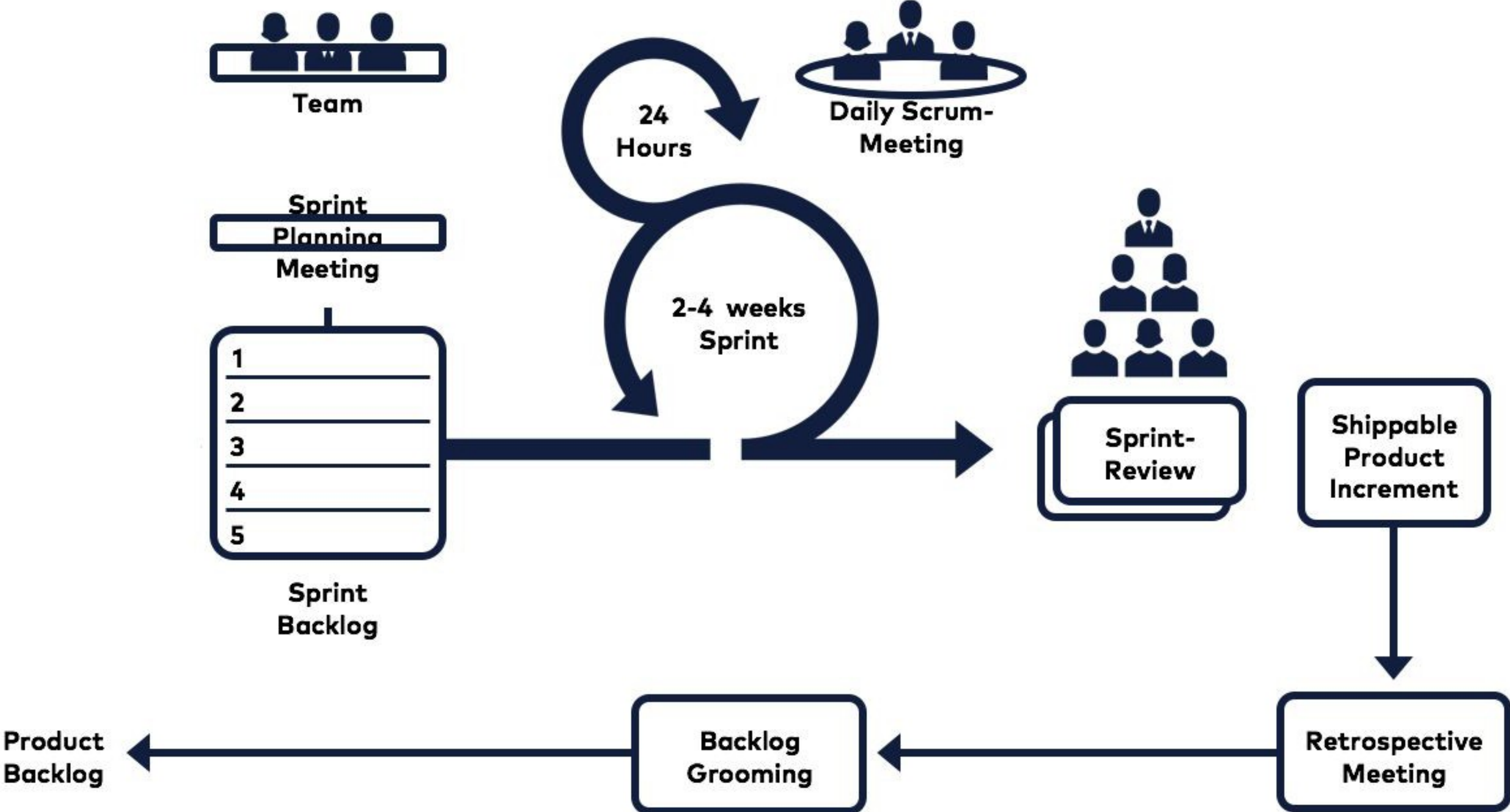
**5 Optimize**  
Continually measuring and optimizing to ensure the best possible experience and results.



# How to Agile

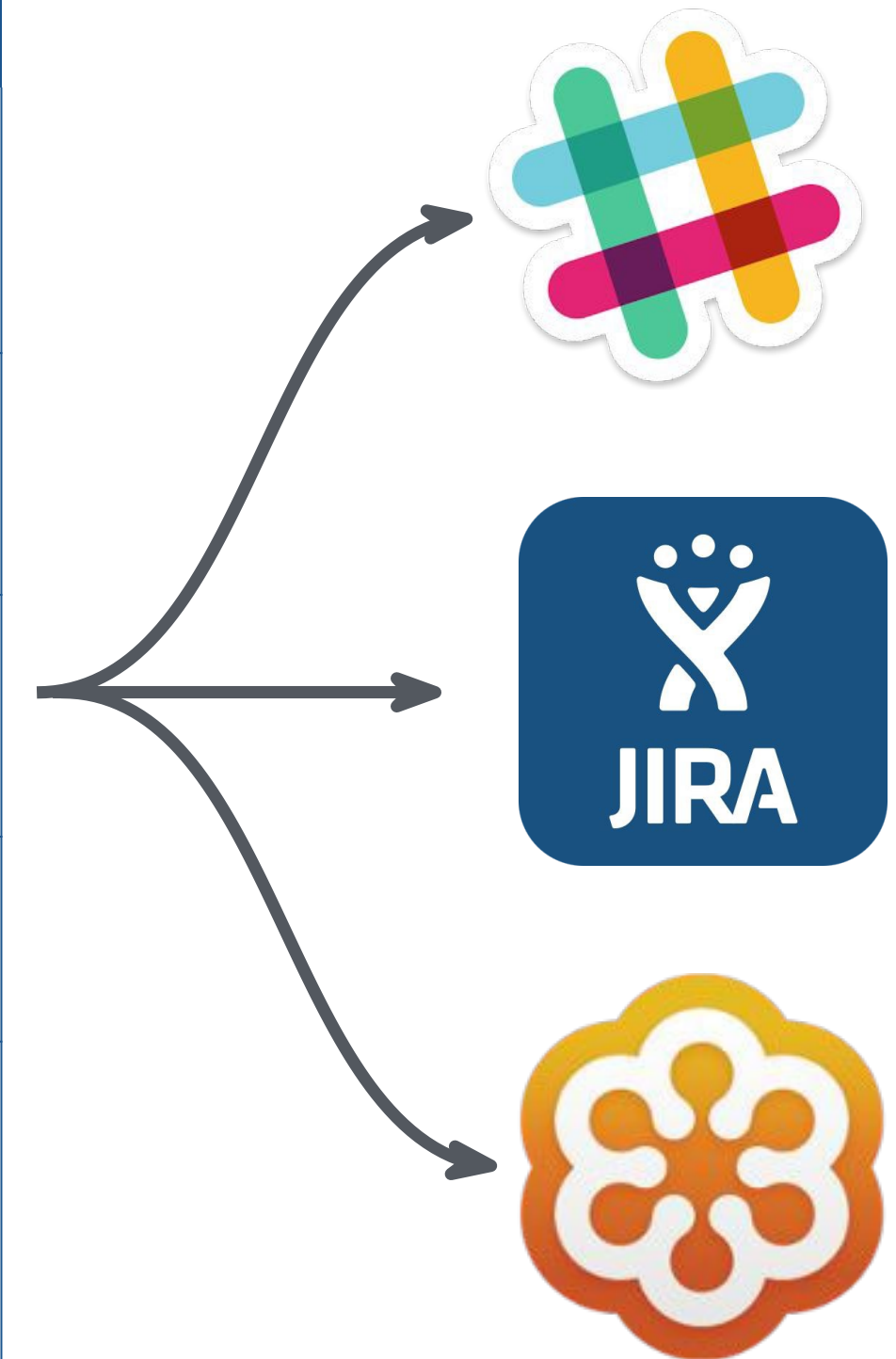
- Tickets are executed using **MoSCoW method of prioritization**.
  - ◆ **MUST** have this requirement to meet the business needs.
  - ◆ **SHOULD** have this requirement if at all possible, but project success does not rely on this.
  - ◆ **COULD** have this requirement if it does not affect the fitness of business needs of the project.
  - ◆ **WON'T** include this requirement in a given release, but it may be considered for the future.
- Agile development runs in **2- to 4-week sprint cycles**.
  - ◆ **Daily standups** keep the team on track and allow blockers to be quickly resolved.
  - ◆ **Project demos** at end of each sprint provide real-time review and course correction to ensure product alignment throughout the development process.

# The Agile Workflow



# Communicating Agile

Activity	Daily	Weekly	End of Sprint	Participants	Comments
Stand upcalls	X			Core team	The sprint team will meet for 10-15 mins to report progress, discuss current issues, and set priorities for the week.
Basecamp communication / Slack	X			Core team, stakeholders	Information is continuously exchanged over Basecamp and quick questions are answered in real-time on Slack.
Status update meetings		X		Core team, stakeholders	The project team meets weekly to report progress, discuss high-level issues, and set longer term priorities.
Sprint demo			X	Core team, stakeholders	Our will demo the results of each sprint to key stakeholders.
Comprehensive progress report			X	Core team, stakeholders	Our will list all current and completed tasks, and will provide a full overview of the hours worked. The report will be submitted at the end of each print to key stakeholders.



# SCRUM



# Scrum Overview

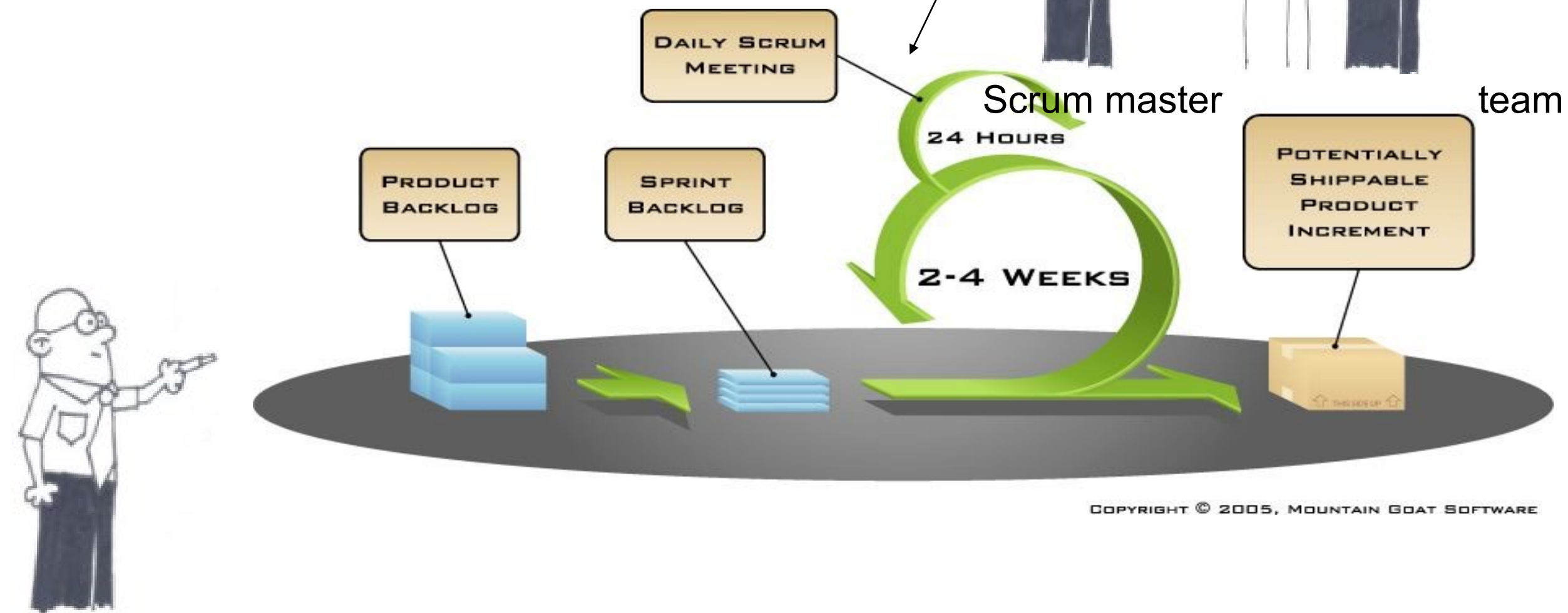
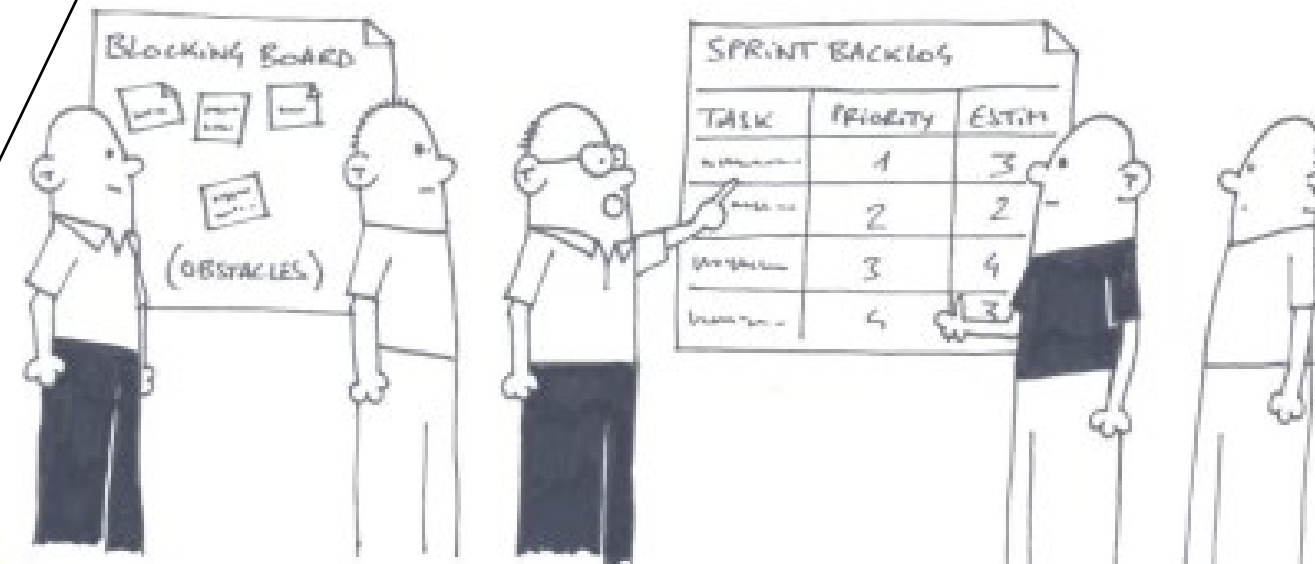
Scrum is an Agile process;  
Used to manage complex projects since 1990;  
Delivers business functionality in 30 days;  
Business sets the priorities;  
Teams self-organize to determine the best way to deliver the highest priority features.  
Scalable to distributed, large, and long projects;  
Extremely simple but very hard!



# Scrum - framework

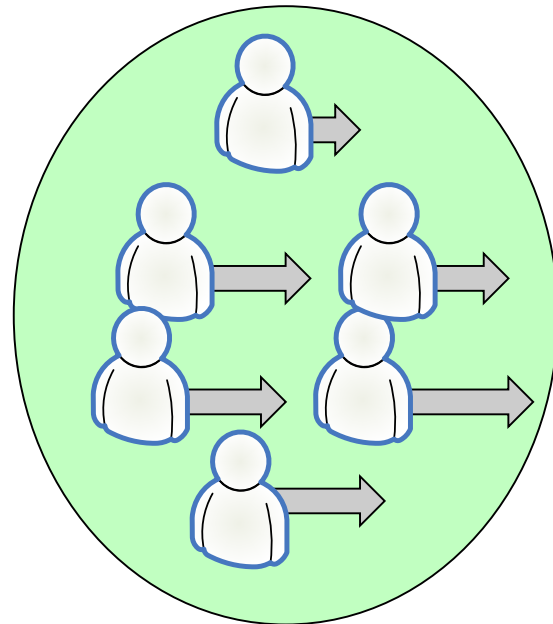
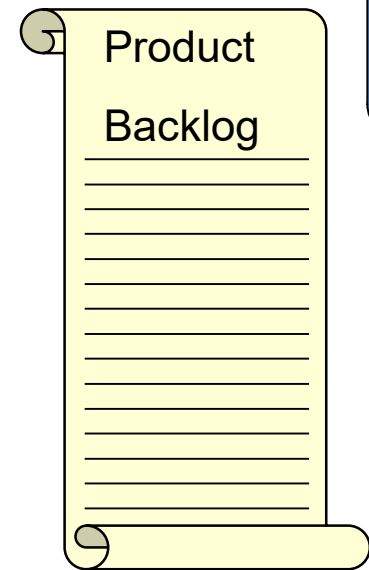
Sprint planning - “definition of Done”  
Sprint review - “the demo”  
Sprint retrospective  
Daily scrum meeting

Timeboxing!



Product owner

# Cross functional team



I *can* test, but I'm not so good at it.

Skills Needed to implement Top X backlog items

	Test	DB	Web	Java	Domain	CM
Lisa	●	●	●	★		●
Joe	●	★		●	●	
Fred	●			★	●	●
Jenny	●		★	●		
David	★		●		★	●
Erik			★	★	●	★

I'm good at Java!

I don't know CM at all. But I'm willing to learn!

I won't even go near a database!

# Team

- Seven (plus/minus two) members
- Is cross-functional (Skills in testing, coding, architecture etc.)
- Selects the Sprint goal and specifies work results
- Has the right to do everything within the boundaries of the project guidelines to reach the Sprint goal
- Organizes itself and its work
- Demos work results to the Product Owner.

# Scrum Master

- Ensure that the team is fully functional and productive
- Enable close cooperation across all roles and functions
- Remove barriers
- Shield the team from external interferences during the Sprint
- Ensure that the process is followed, including issuing invitations to Daily Scrum, Sprint Review and Sprint Planning meetings.

# Product Owner

- Define the features of the product.
- Decide on release date and content.
- Be responsible for the profitability of the product (ROI).
- Prioritize features according to market value.
- Adjust features and priority every iteration, as needed
- Accept or reject work results.

# Scrum Elements

## THREE Roles

- Product Owner
- Scrum Master
- Team Member

## THREE Meetings

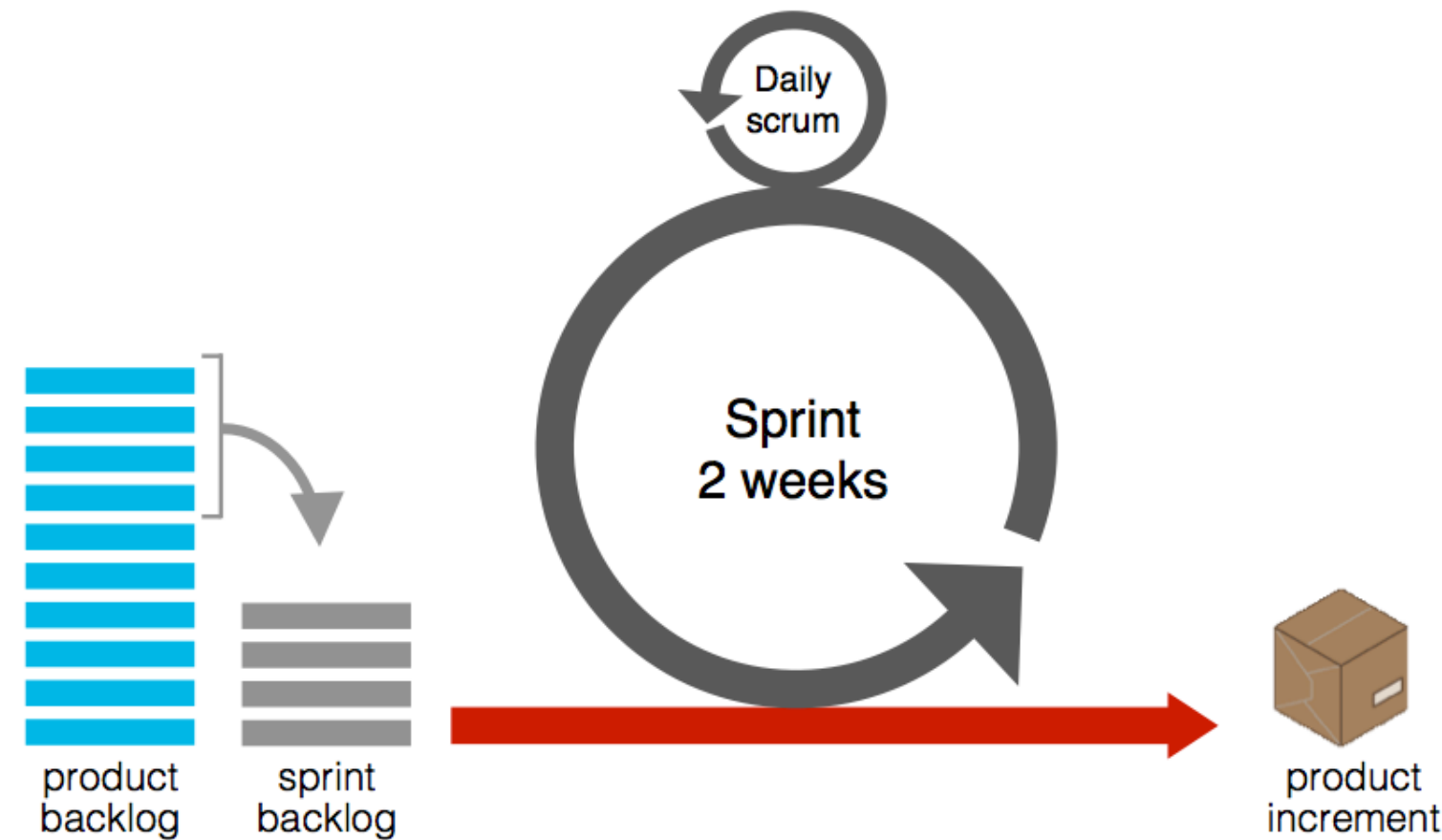
- Planning (Release & Sprint)
- Daily Scrum
- Sprint Review

## THREE Lists

- Product Backlog
- Sprint Backlog
- Impediments List

## Scrum

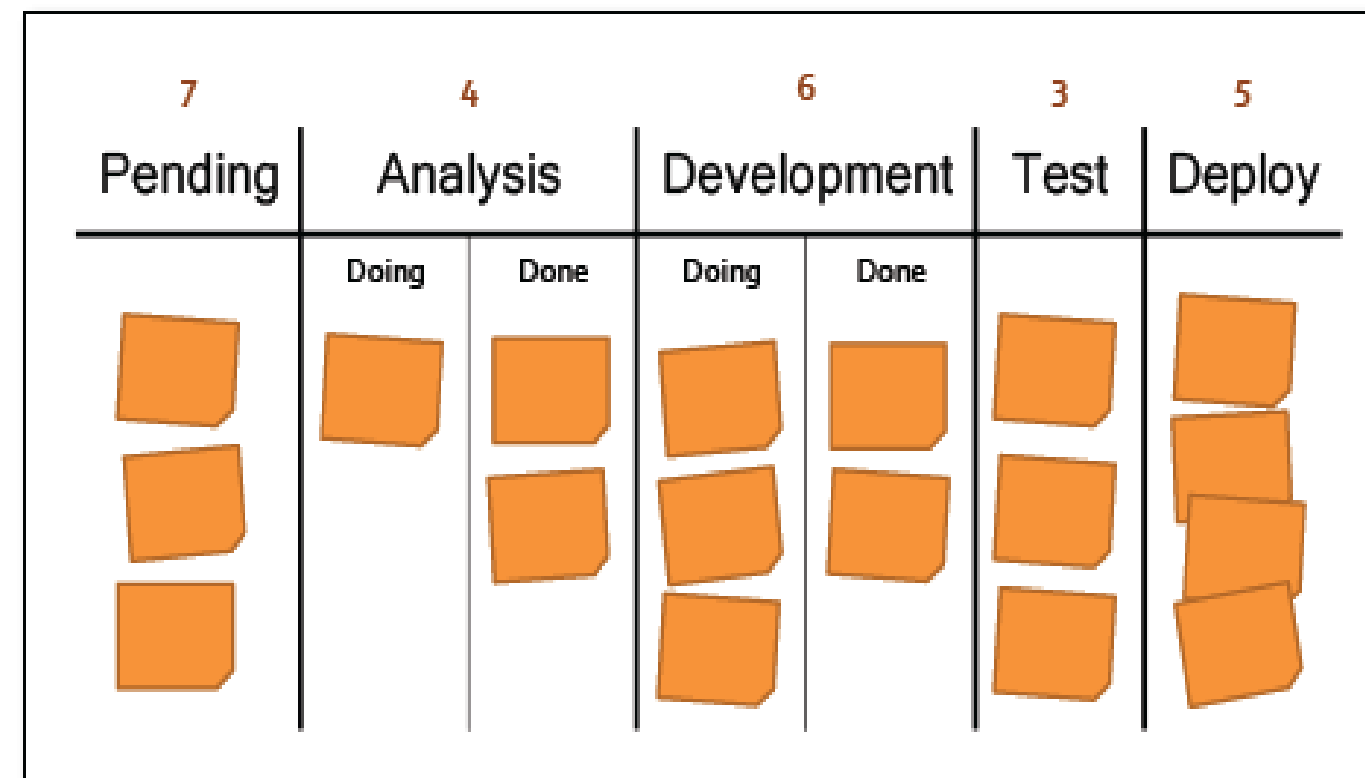
For application Development



Scrum is a simple yet incredibly powerful set of principles and practices that help teams deliver products in short cycles, enabling fast feedback, continual improvement, and rapid adaptation to change. As the leading Agile development framework, Scrum has predominantly been used for software development, but it is also proving to be effective in efforts far beyond.

## Kanban

For Infrastructure, Portfolio Management, and Business Change



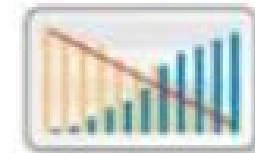
Kanban is a continuous flow process: items enter the queue and then get “pulled” through a series of steps in the development process. Kanban is often visualized on a Kanban board and each step is represented by a column.

# The Agile: Scrum Framework at a glance

Inputs from Executives,  
Team, Stakeholders,  
Customers, Users



Burndown/up  
Charts



Every  
24 Hours



Sprint end date and  
team deliverable  
do not change





## Enterprise Portfolio Management Office

### Portfolio Management (Build IT Services)

- Portfolio Management
- Program Management
- Project Management
- Roadmapping
- Release Planning

### Service Management (Run IT Services)

- Service Desk
- Service Level Management
- Availability Management
- Security Management
- Service Continuity Management
- Incident Management
- Problem Management
- Configuration Management
- Change Management

### Business Change Management (Manage Change)

- Business Outcomes/KPIs
- Business Process Management
- Organizational Change
- Business Rules
- Business Information Management

# Software and Services for Powering Business Value

## **Enfocus Solutions can be applied to a variety of areas including:**

End-to-End Business and IT Service Design including creating and maintaining SDPs

Agile Portfolio and Program Management

Collaborative Business Architecture

Feature discovery, prioritization and validation

Managing business change (Impacts)

Performing Business Analysis as defined in BABOK

Defining traditional requirements for evaluating COTS and Cloud Solutions

## **Enfocus Solutions provides the following benefits**

Achieve better business outcomes and higher ROI on Projects

Enables agile to scale to the Enterprise

Provides business transparency and enables engagement of stakeholders

Reduce costs by removing wastes from IT services and value streams

Area	Enfocus Capabilities	Business Outcomes
Agile Development	<ul style="list-style-type: none"> <li>• Integrate with Team Tools such as JIRA</li> <li>• Define and Validate Features</li> <li>• User Stories</li> <li>• Manage backlog for multiple teams</li> <li>• Non-functional Requirements</li> <li>• Test scenarios and Test case</li> </ul>	<ul style="list-style-type: none"> <li>• Higher quality software</li> <li>• Allows agile to scale</li> </ul>
Portfolio and Program Management	<ul style="list-style-type: none"> <li>• Support for Scaled Agile Framework</li> <li>• Roadmapping (Coming)</li> <li>• Release Planning</li> <li>• Inspect and Adapt</li> <li>• Feature and Validation</li> <li>• Transparency and Collaboration</li> </ul>	<ul style="list-style-type: none"> <li>• Shorter Cycle Times</li> <li>• More delivery of value</li> <li>• Coordination of cross-functional teams</li> <li>• Lower costs through elimination of low value work</li> </ul>
Service Strategy and Design	<ul style="list-style-type: none"> <li>• Service Portfolio Management</li> <li>• Service Design Packages</li> <li>• Integrated Business Architecture</li> </ul>	<ul style="list-style-type: none"> <li>• Better customer experience</li> </ul>
Release and Deployment Management	<ul style="list-style-type: none"> <li>• Transition Requirements</li> <li>• Inspect and Adapt</li> <li>• Release Management</li> </ul>	<ul style="list-style-type: none"> <li>• Increased transparency</li> </ul>
Lean Value Streams	<ul style="list-style-type: none"> <li>• Lifecycle management</li> <li>• Metrics and measurement</li> </ul>	<ul style="list-style-type: none"> <li>• Less Waste</li> <li>• Shorter cycle times</li> </ul>
Lean Business Change Management	<ul style="list-style-type: none"> <li>• Understand impacts on people, technology, processes, and data</li> <li>• Provide transparency to business stakeholders</li> </ul>	<ul style="list-style-type: none"> <li>• Faster user adoption</li> <li>• Shorter time to value</li> </ul>