Towards the .NET Junior Developer

The extremely solid course

Lesson 12

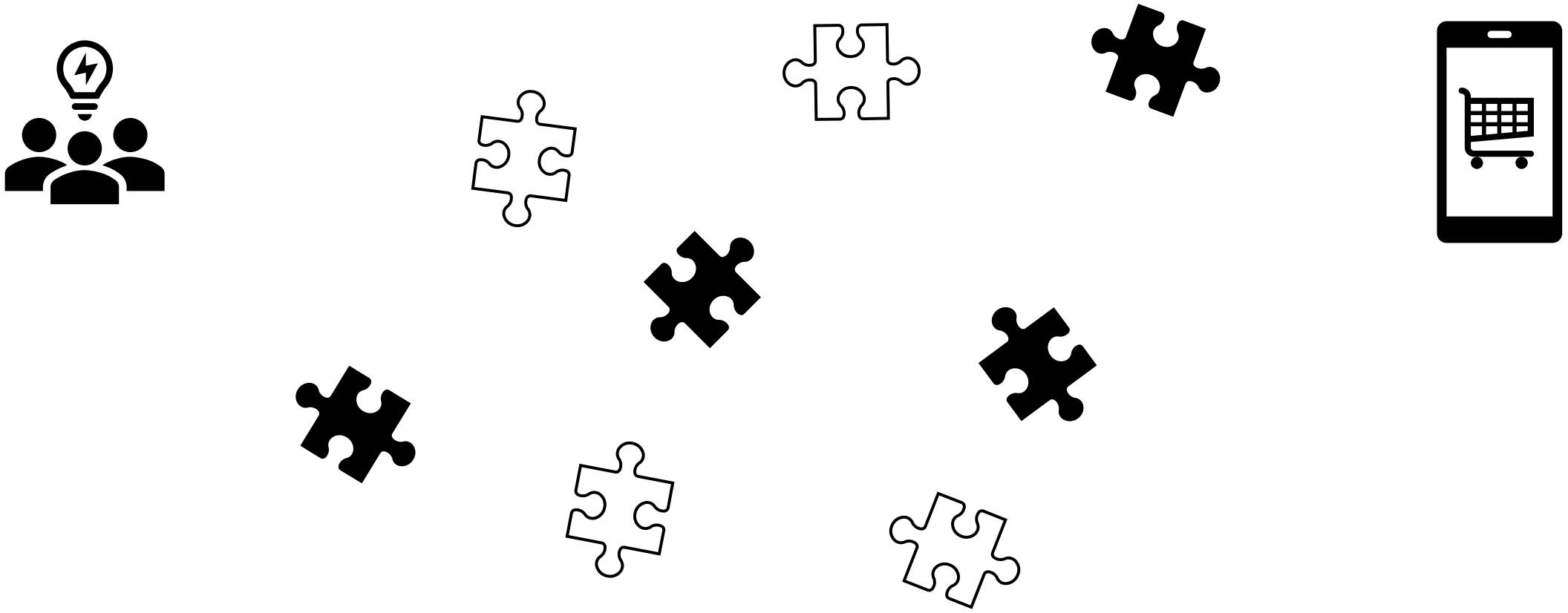
Team work

Agenda

- Software development process basics
  - Business analysis
  - Architecture design
  - Development
  - Code review
  - Testing
  - Deployment
  - Support
- Software development methodologies
  - Waterfall
  - Agile
    - Kanban
    - Scrum
- Books of the day
- Links of the day
- Hometask

Software development process

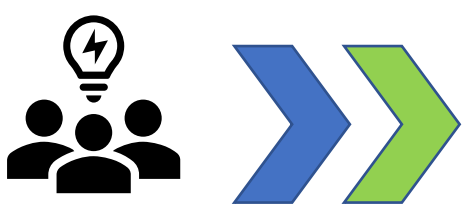
Software development process



Business analysis

- Collect primary requirements
- Fix use cases
- Clarify the requirements for the developers
- Discuss product details with the customers
- Write technical specification (sometimes)

Architecture design



- Analyze the requirements
- Make decision about ability and reasonability of the changes/development
- Prepare hi-level design (especially for new projects)
- Choose technological stack
- Create Proof Of Concept (PoC) (if needed)
- Work with senior developers in the context of the system design and approaches

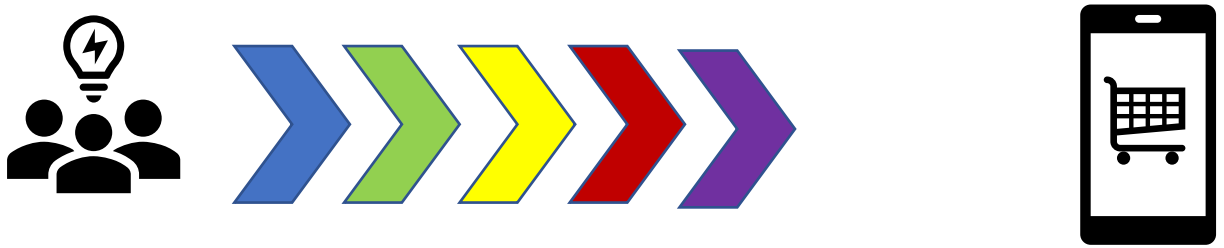
Development

- Analyze the requirements and architecture design, estimate work
- Communicate with BA and architect
- Propose the most effective low-level implementation of the requirements
- Investigate new approaches and best practices
- Write and test code
- Participate in the testing, deployment and support stages

Code review

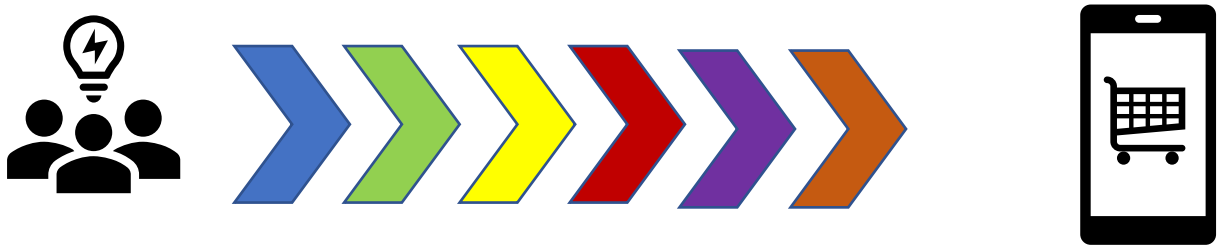
- Analyze changes in terms of quality, correctness, code conventions, ability to support
- Propose the problem solutions or more effective approaches
- Pass code on the colleague’s analysis
- Participate in discussions about changes
- Fix found problems and non-optimal approaches (duplication, incorrect logic etc.)

Testing



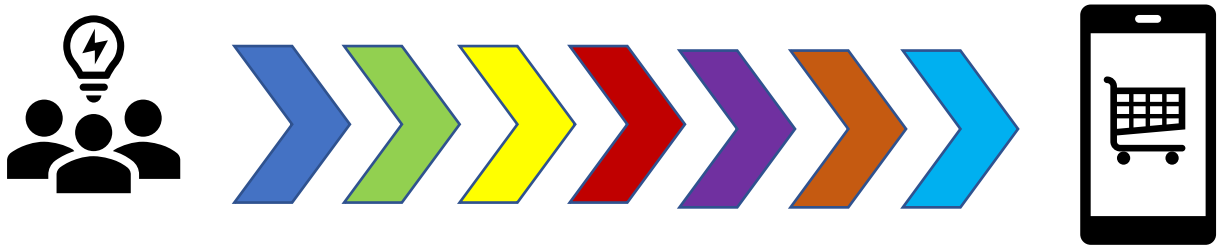
- Describe new scenarios (use cases)
- Pass through the known scenarios and validations
- Write auto-tests (for auto-testers only)
- Create the bug tickets for the developers to fix bugs and issues
- Communicate with the developers and analysts to describe the found issues and bugs

Deployment



- Prepare and setup infrastructure
- Describe deployment pipelines
- Implement pipelines in the CI/CD systems
- Communicate with the developers in part of deployment and pipelines details
- Watch for the pipeline's correctness and workability

Support



- Watch for the system/application/service’s metrics and logs
- Fix found issues
- Work with the technical debt
- Initialize/develop/implement new features and updates
- Communicate with the customers and end users

Software development methodologies

Waterfall

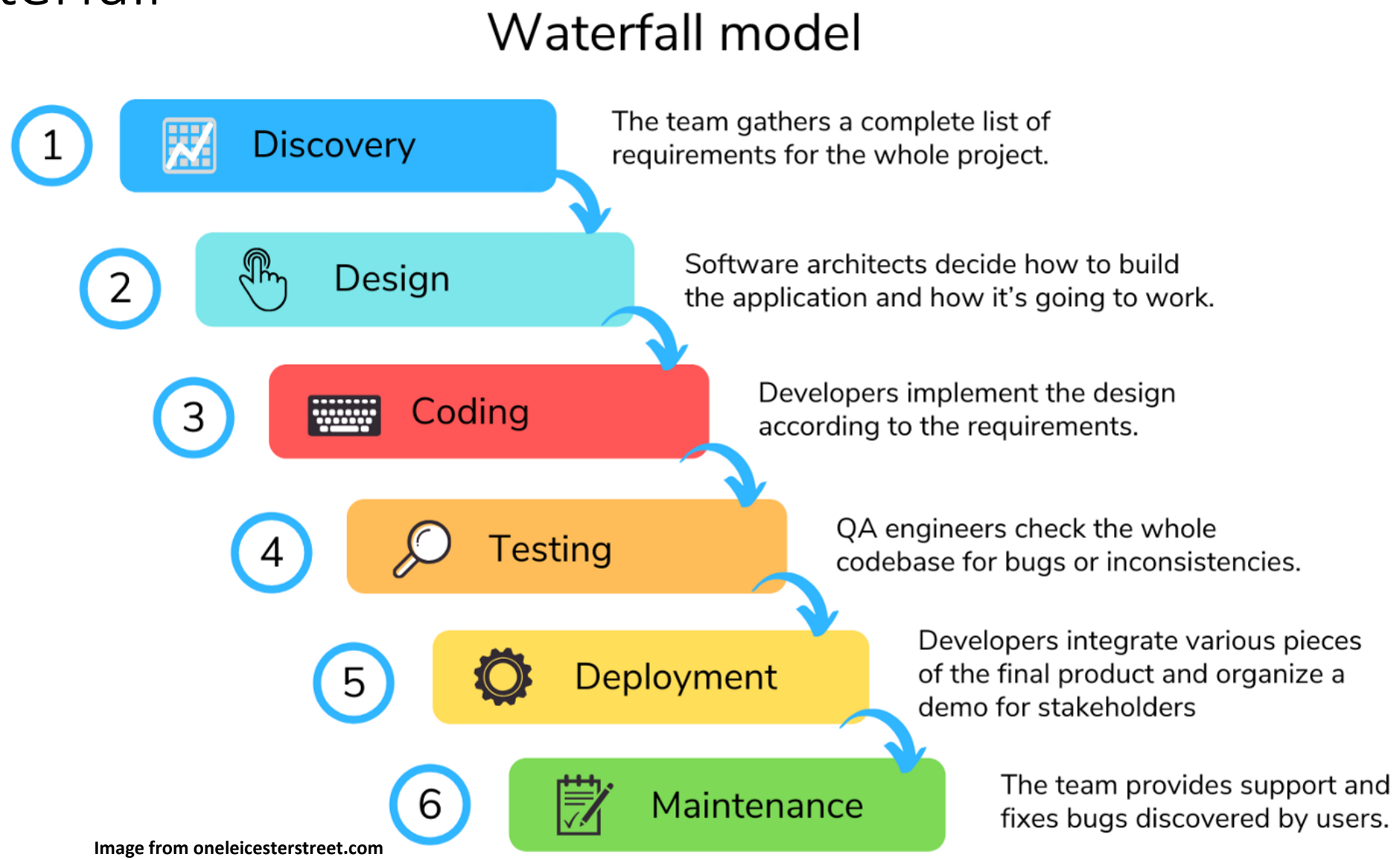


Image from oneleicesterstreet.com

Agile

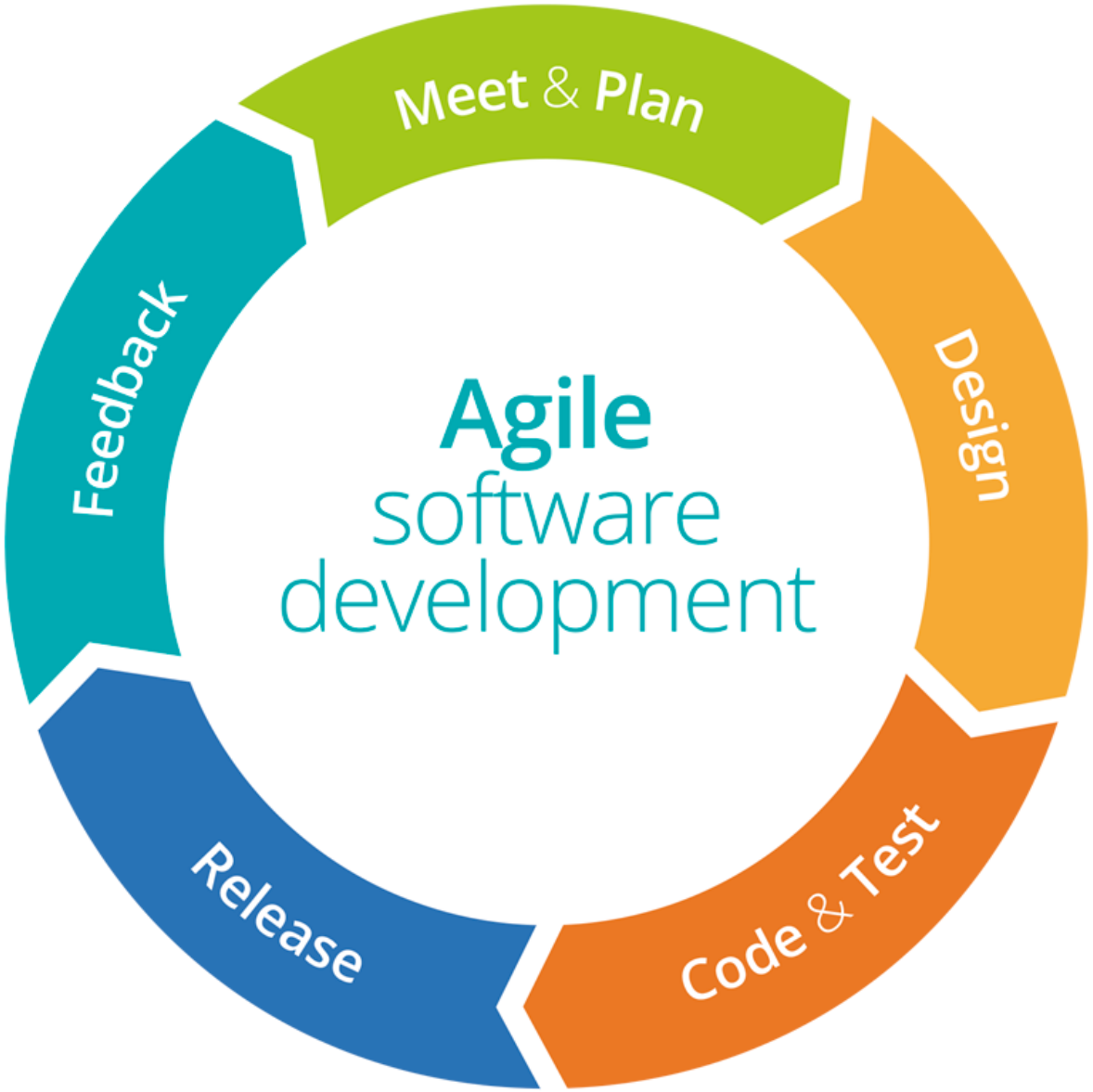


Image from medium.com

Agile

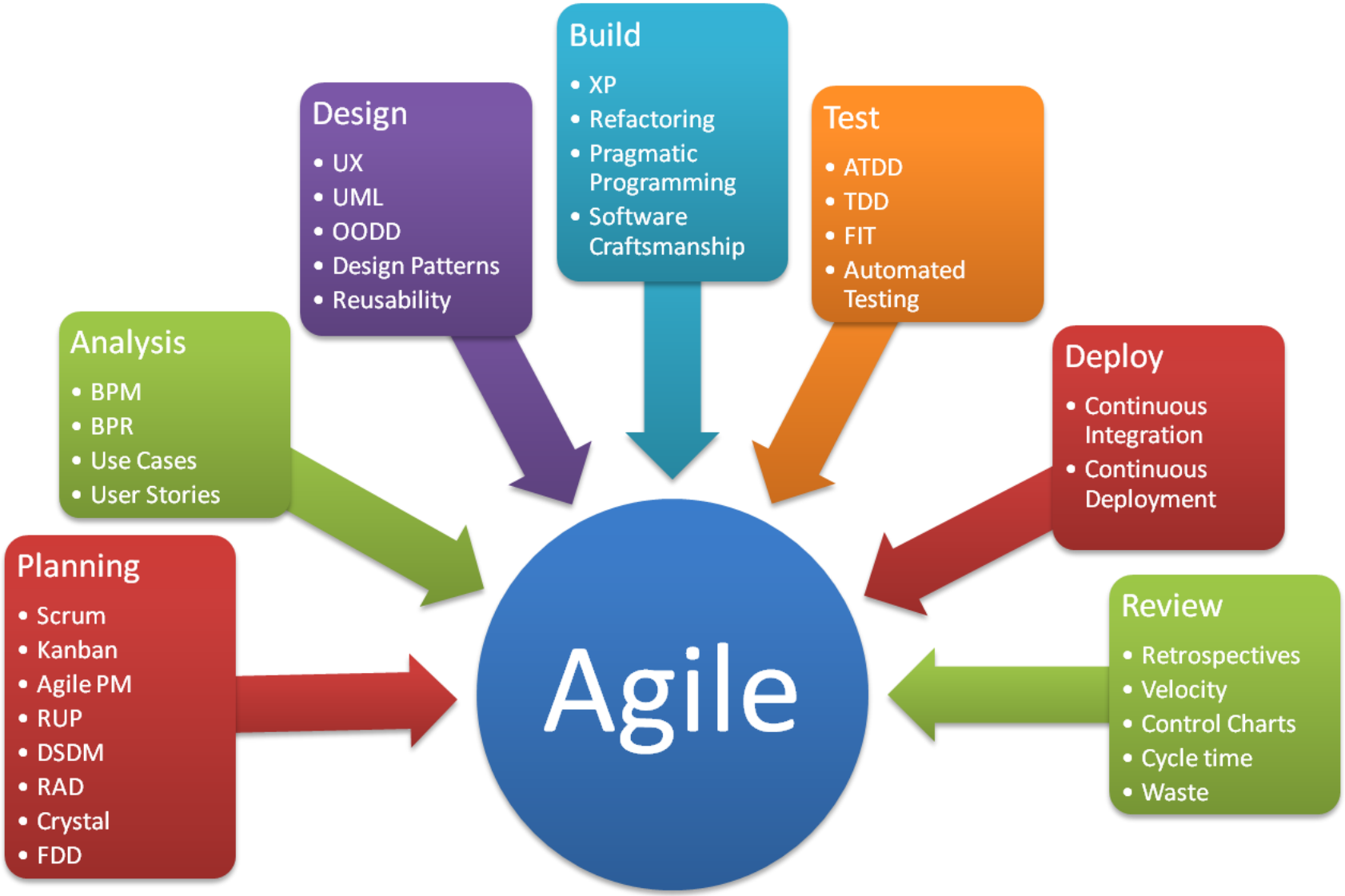
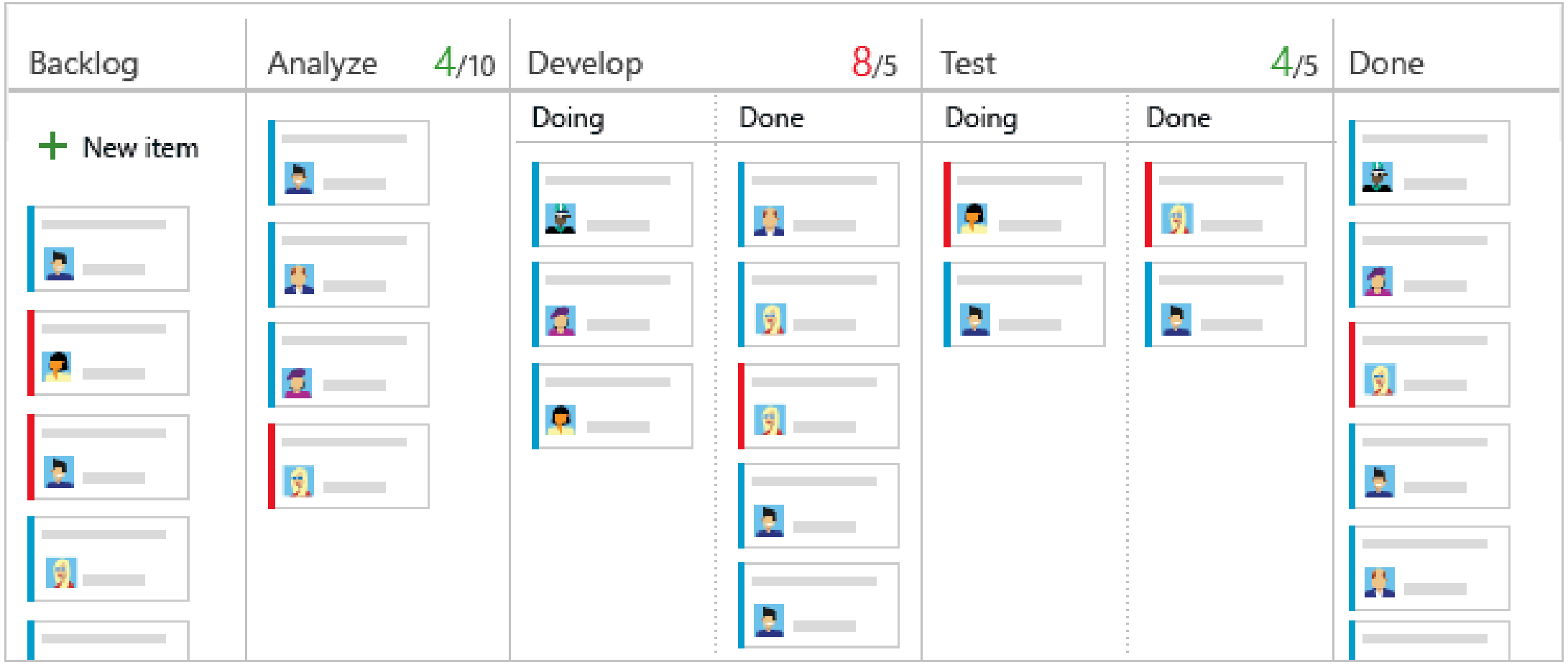


Image from pinterest.com

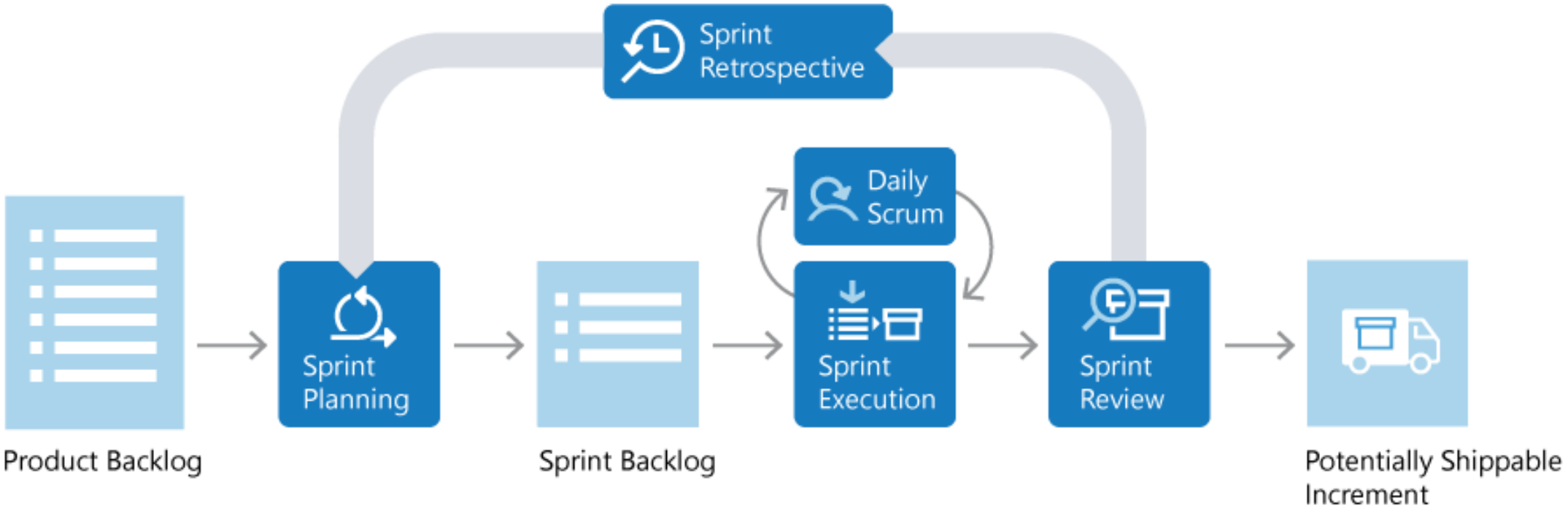
Agile

- **Individuals** and **interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

Kanban


Backlog
Analyze 4/10
Develop 8/5
Test 4/5
Done
+ New item
Doing
Done
Doing
Done

Scrum


Sprint Retrospective
Daily Scrum
Sprint Planning
Sprint Execution
Sprint Review
Product Backlog
Sprint Backlog
Potentially Shippable Increment

Scrum

Artifacts:

- Product Vision
- Sprint Goal
- Product Backlog
- Sprint Backlog
- Definition of Done
- Burn-Down Chart
- Increment

Roles:

- Stakeholder
- Product Owner
- Scrum Master
- Scrum Team

Books of the day

McConnell S. – Code Complete

Stellman A. – Learning Agile

Martin R. – Clean Agile

Martin R. – Clean Craftsmanship

Links of the day

[Software Development Process: The Ultimate Guide (syndicode.com)]

[Manifesto for Agile Software Development (agilemanifesto.org)]

[What is Scrum? - Azure DevOps | Microsoft Docs]

[Roles in Scrum Methodology (hygger.io)]

[Методология Kanban: введение / Хабр (habr.com)]

[Где Agile ужасен, особенно Scrum / Хабр (habr.com)]

Hometask

Work on your project!

That’s all for this time!