Software Project Management - Laboratory

Lecture n° 1
A.Y. 2020-2021

Who is Fabrizio Fornari?

2020 Postdoc in Computer Science at UNICAM

2018 PhD title in Computer Science at UNICAM. 3 months in Brisbane Queensland University of Technology (Australia)

2012-2013 Master’s degree in Computer Science at UNICAM and University of Reykjavik (Iceland)

2010-2011 Bachelor degree in Computer Science at UNICAM

Collaboration

PROS Lab - PROcesses and Services Laboratory http://pros.unicam.it/

Past collaborations:

- National Research Center of Pisa (CNR)
- Apromore group https://apromore.org/ (University of Melbourne)
- Technical University of Denmark
- Sant’Anna School of Advanced Studies (Pisa)

PROS Lab
pros.unicam.it

My Research Topics

Business Process Management
Business Process Modeling and Verification
Business Process and IoT

Tool:

- BProVe, Business Process Verifier
- BEBoP, understandaBility vErifier for Business Process models
- RePROSitory, Repository of open PROcess modelS

Teaching

2020-2021

- Software Project Management Laboratory at “University of Camerino”, Department of Computer Science (6CFU)

2019-2020

- Computer Science at “Università di Macerata”, Faculty of “Economia e diritto” (6CFU)

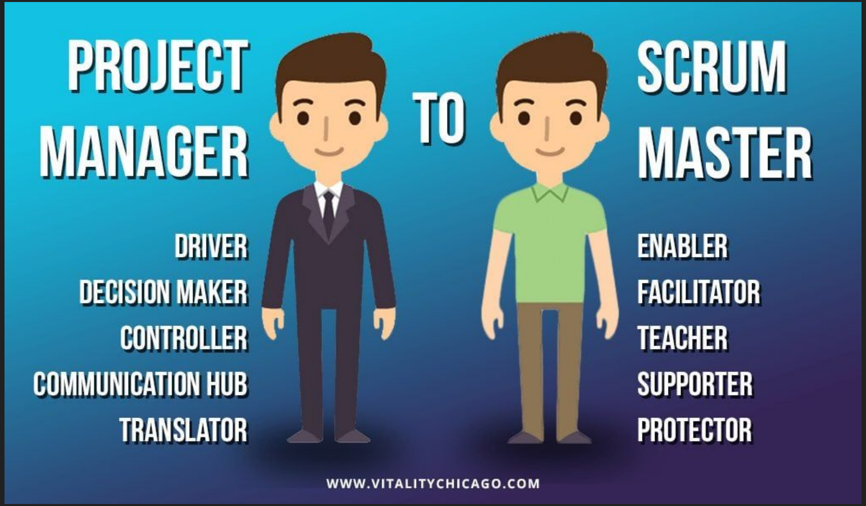
2018-2019

- Computer Science at “Università di Macerata”, Faculty of “Economia e diritto” (6CFU)
- Software Project Management Laboratory at “University of Camerino”, Department of Computer Science (3CFU)

Group Projects

I supervised/co-supervised a dozen of group projects and experimental thesis.

I try to apply together with the students the methodology that we will see during the course.



Course Overview

Course Objective

The course introduces the students to the basic knowledge of complex software system production following the **DevOps methodology**.

Prerequisite knowledge

- Basic Programming Experience
- Basic Software Engineering Methods and Techniques

Learning Outcome

The student will be able to manage the organization and the development of a software applying DevOps methodology.

Course Overview

Website:

- http://didattica.cs.unicam.it/doku.php?id=didattica:magistrale:spm:ay_2021:main&s[]=spm

Teaching Hours:

- Thursday 9am-11am (Lab)
- Friday 11am - 1pm (Lab)

Students Meeting:

- After each lesson or,
- By requesting a meeting by sending an email to *fabrizio.fornari@unicam.it*

Note: only email coming from the @studenti.unicam.it domain will be processed.

Software Requirements

- Java (type java -version in a terminal/command line)
- Eclipse IDE for Java Developers (or an IDE of your choice)

Exam

The exam involves the development of a software project by following the methodologies introduced during lecture hours with the objective of supporting a “Smart Parking” system.

Exam Evaluation:

- Git Usage
- Testing
- SCRUM application/Sprint management
- Presentation
- Overall Project

Group Formation

- Group of 2 or 3 students have to be formed
- A Project will be assigned
- A GitHub account is needed for every student
- A GitHub repository will be assigned to each group

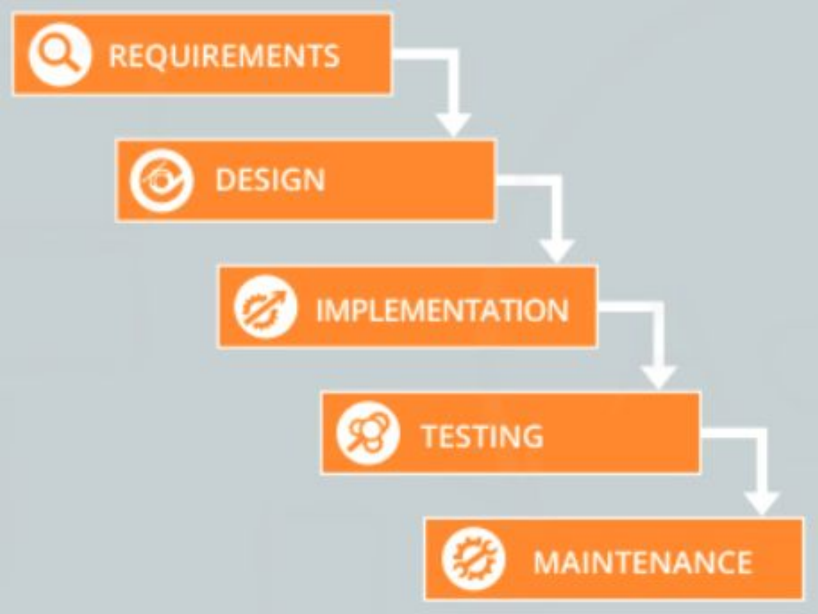
What is a Software Development Process?

Software Development Process

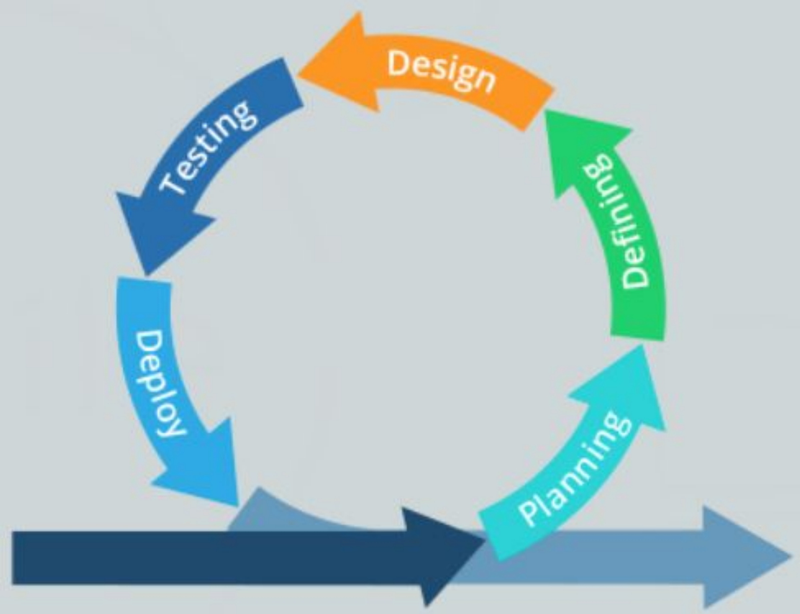
Software Development Process is the process of dividing software development work into distinct phases to improve design, product management, and project management. It is also known as a software development life cycle (SDLC)

Software Development Process

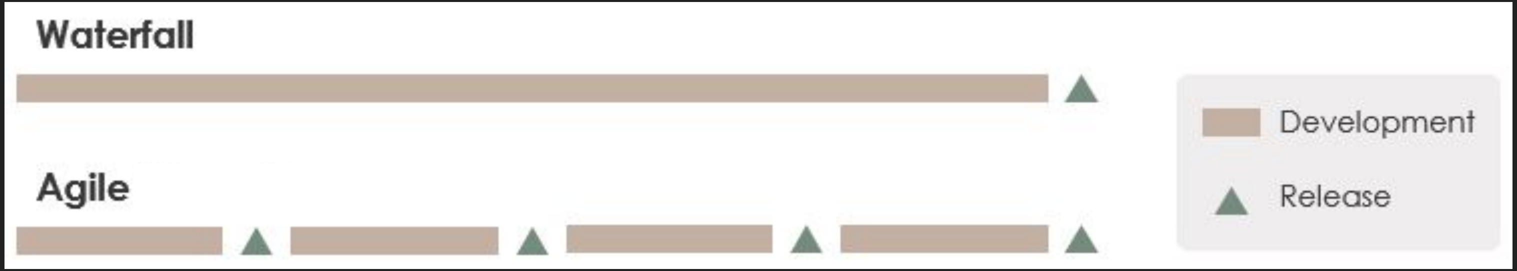
Waterfall



Agile



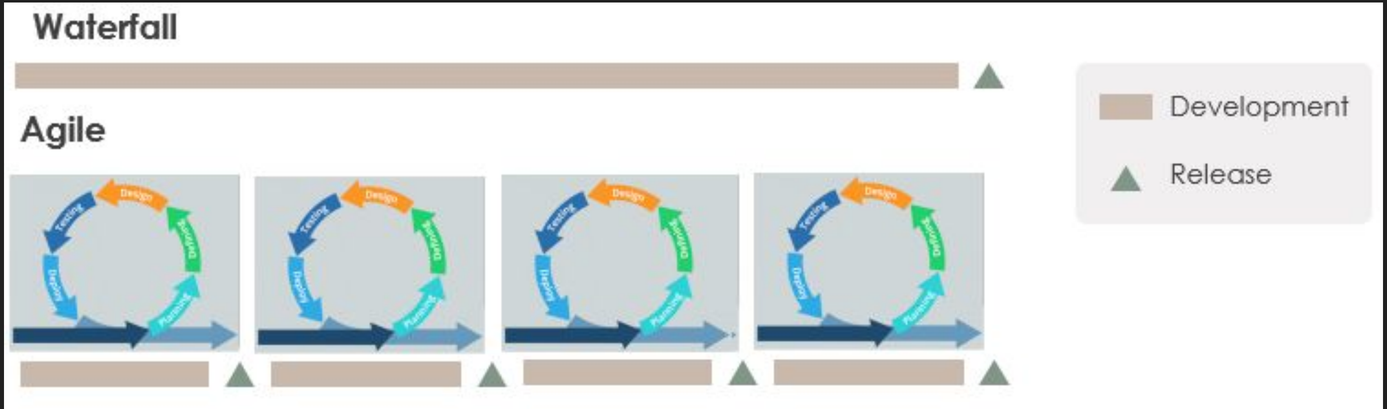
Waterfall vs Agile



Agile characteristics:

- Complex work is divided in simple pieces
- Large organizations are divided into small teams
- Far-reaching project are divided into short time lists of task called sprints

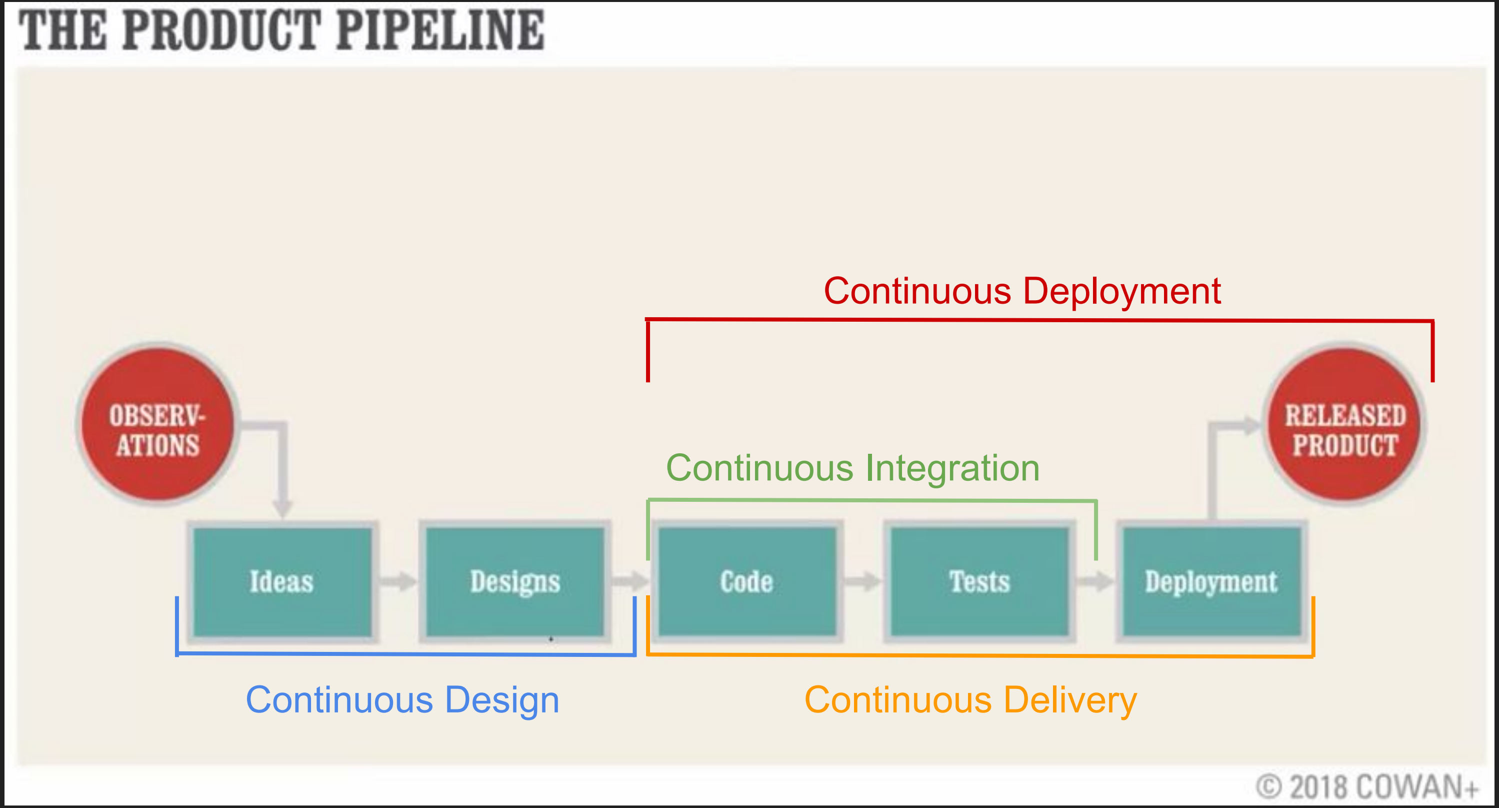
Waterfall vs Agile



Agile characteristics:

- Complex work is divided in simple pieces
- Large organizations are divided into small teams
- Far-reaching project are divided into short time lists of task called sprints

The Product Pipeline



Roles & Interfaces

DEVELOPER

Inputs: User stories

Outputs: Software design & Implementation

TESTER

Inputs: Working software, notes on target behavior

Outputs: Validated software

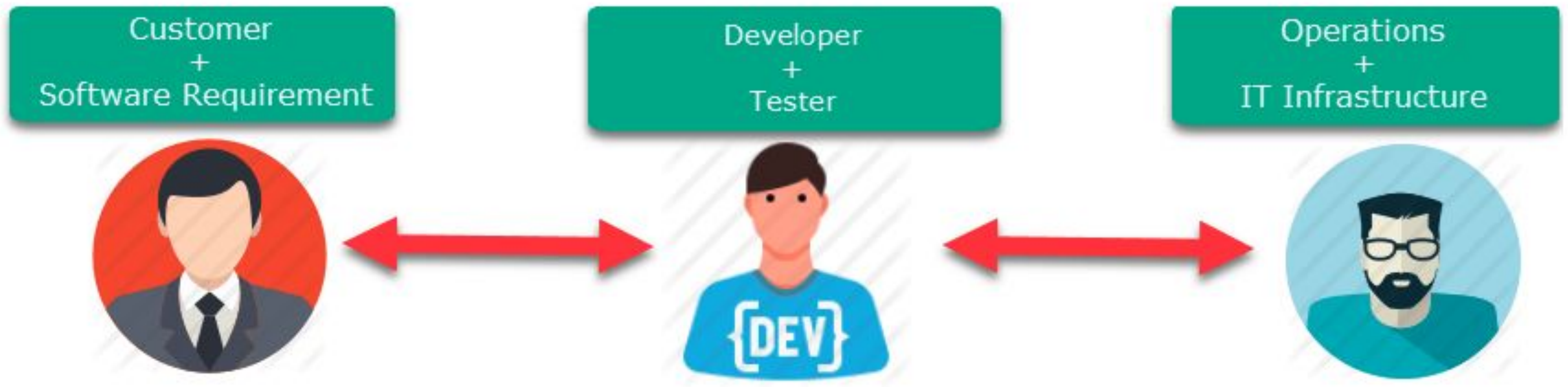
OPERATORS

Inputs: Validated software deployment notes

Outputs: Working systems, monitoring & analytics thereof   evaluates what the software is doing and if it behaves as expected

Classic (and Old) Process

Stakeholders and communication chain in a typical IT process.

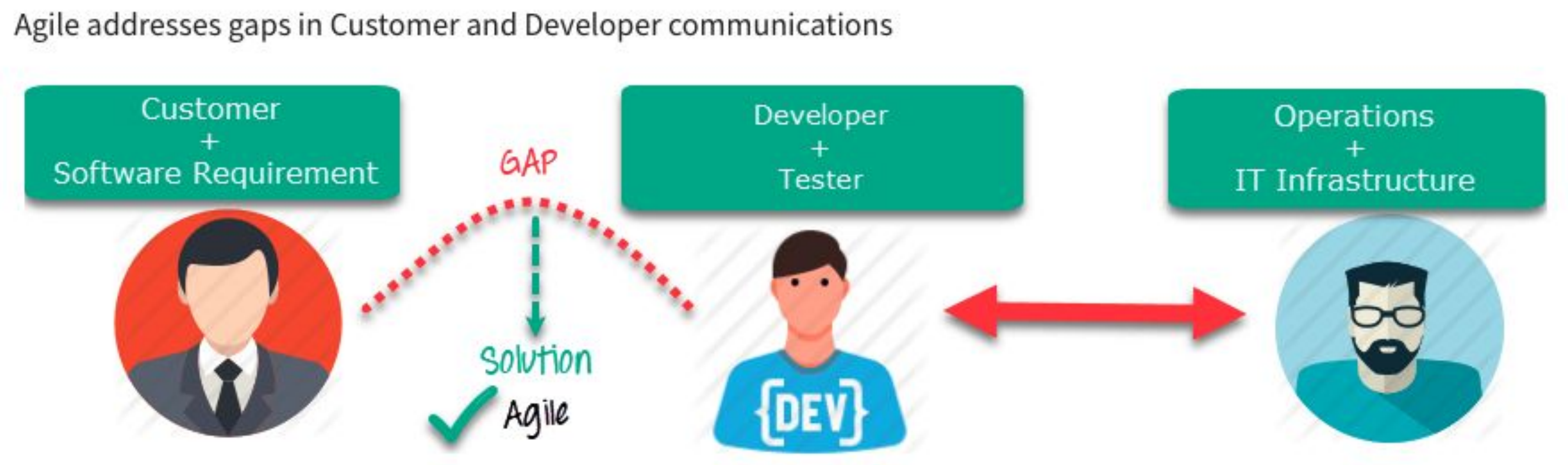


The Agile Manifesto

Individuals Interactions	>	Processes Tools
Working Software	>	Comprehensive Documentation
Customer Collaboration	>	Contract Negotiation
Responding to Change	>	Following a Plan

https://agilemanifesto.org/

Agile



Agile addresses gaps in Customer and Developer communications

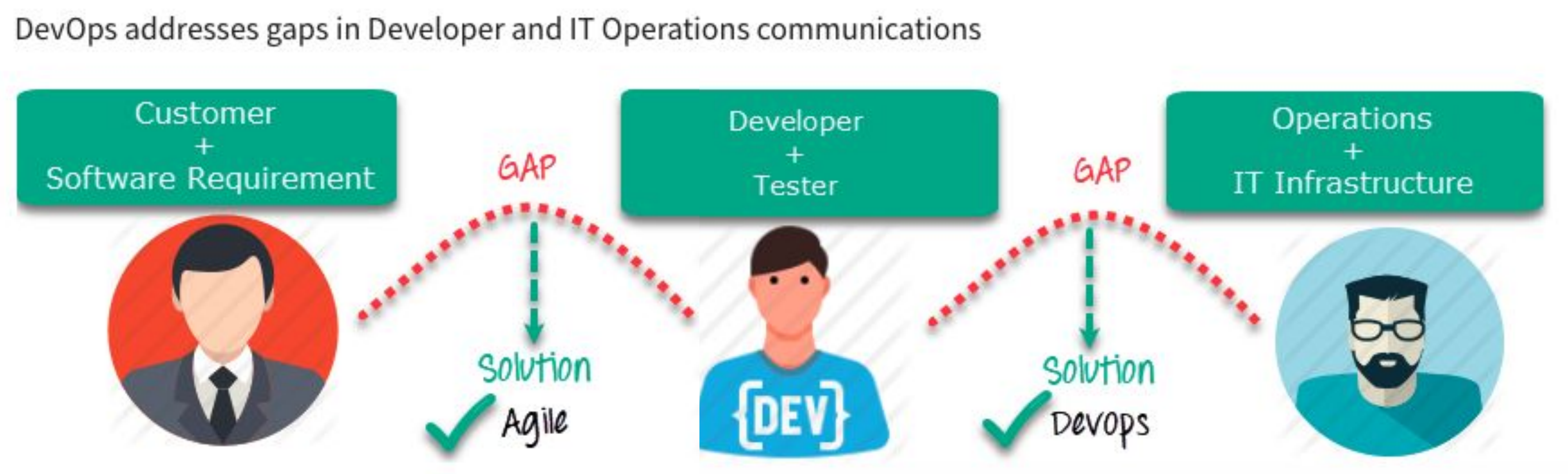
DevOps

“A compound of development (Dev) and operations (Ops), DevOps is the union of people, process, and technology to continually provide value to customers.

What does DevOps mean for teams? DevOps enables formerly siloed roles — development, IT operations, quality engineering, and security — to coordinate and collaborate to produce better, more reliable products. By adopting a DevOps culture along with DevOps practices and tools, teams gain the ability to better respond to customer needs, increase confidence in the applications they build, and achieve business goals faster.”

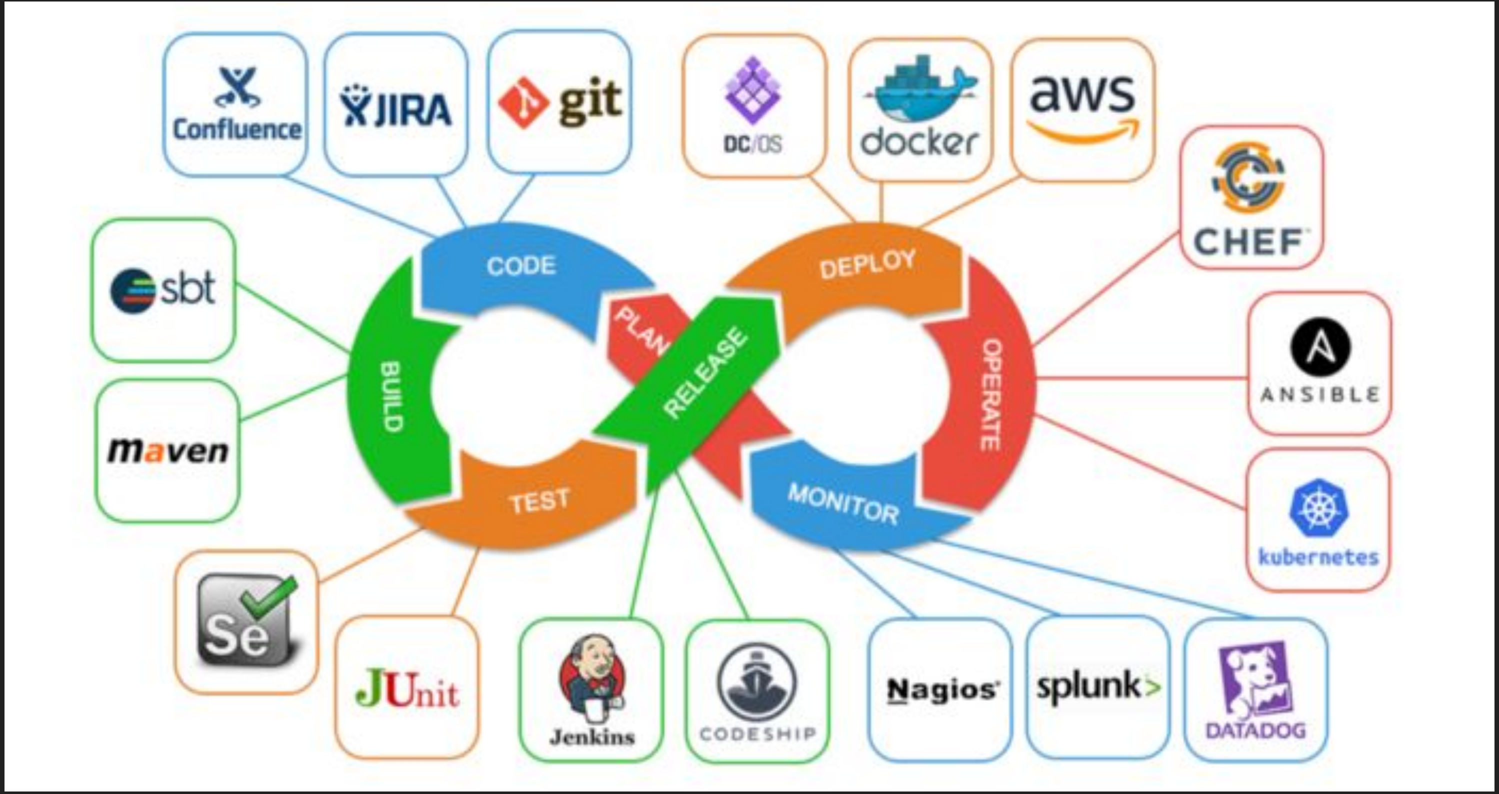
— Azure.microsoft.com

DevOps

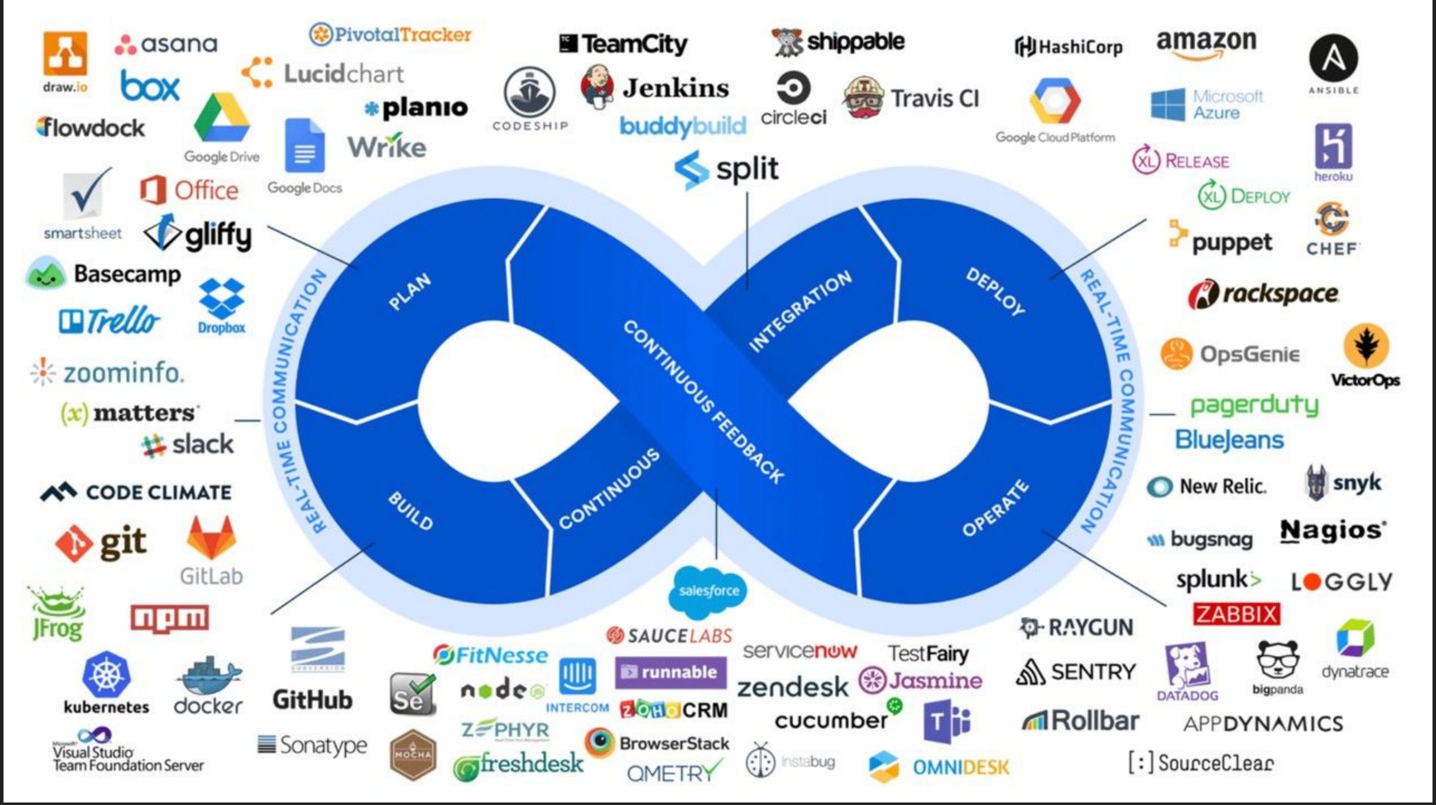


DevOps addresses gaps in Developer and IT Operations communications

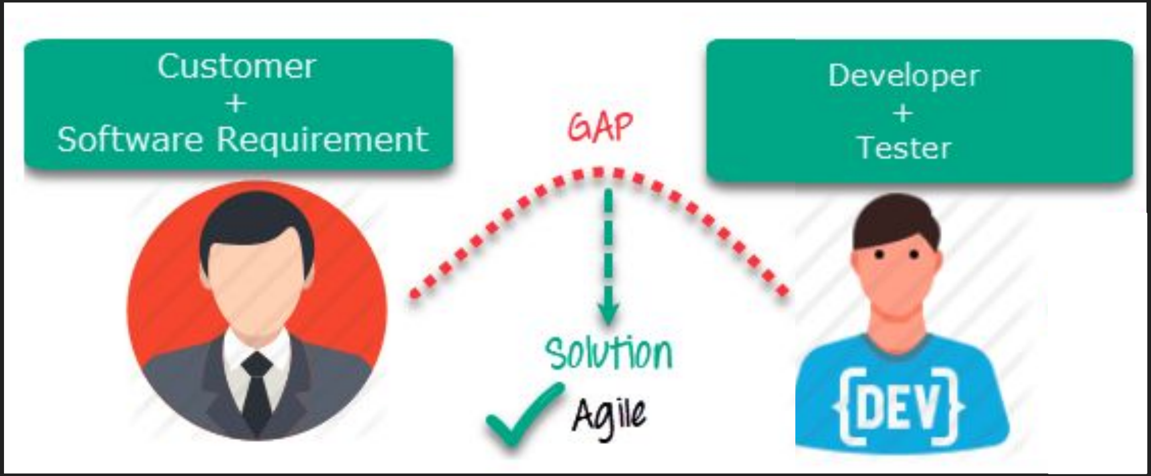
DevOps Technologies



DevOps Technologies

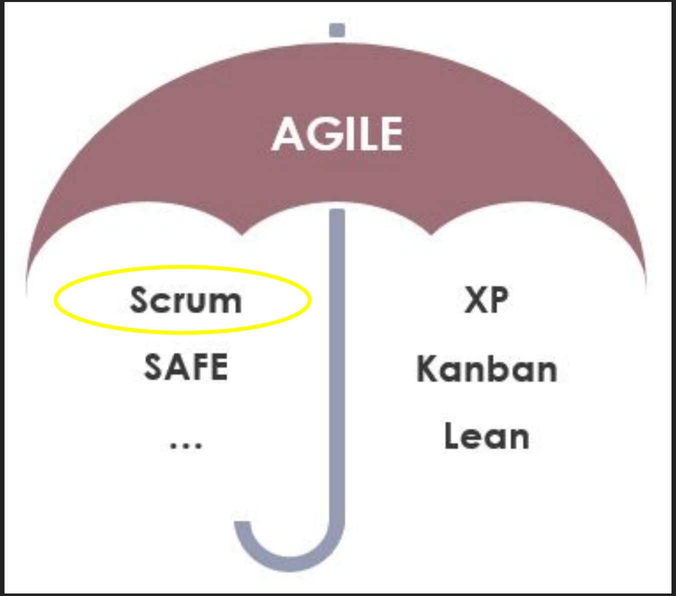


Let’s focus on the Agile part



SCRUM

Scrum is an Agile framework for project management that emphasizes teamwork, accountability and iterative progress toward a well-defined goal.



SCRUM

The Scrum Framework deals with the fact that the requirements are likely to change or not known at the start of the project.

Scrum is:

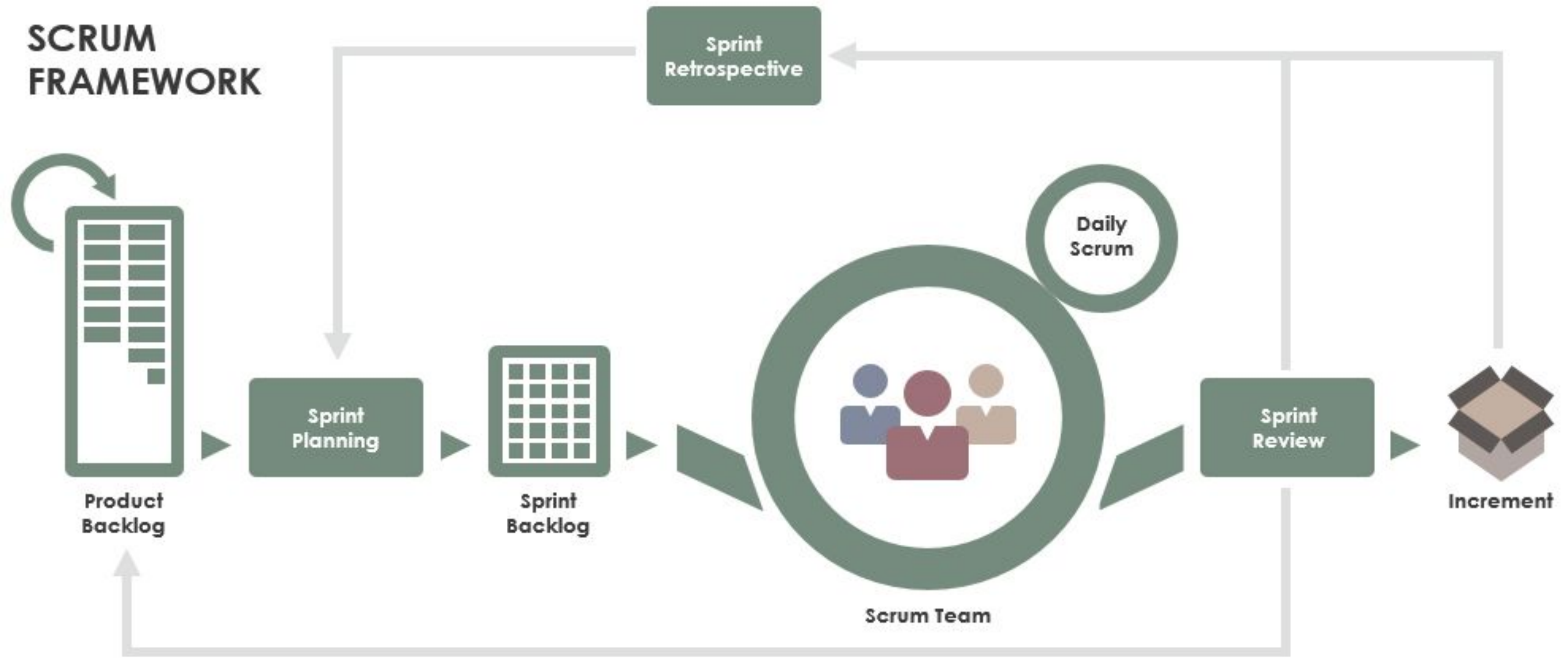
- Lightweight
- Simple to understand
- Difficult to master

SCRUM

Benefits:

- Better Quality Products
- Reduced time to market
- Increase Return on Investment
- Higher Team Morale
- Enhance Team Collaboration

SCRUM


SCRUM FRAMEWORK
Sprint Retrospective
Daily Scrum
Sprint Planning
Sprint Review
Product Backlog
Sprint Backlog
Increment
Scrum Team

SCRUM

The main components of the Scrum Framework are:

- **Roles**
- **Artifacts**
- **Events**
- **Sprint**

Roles

Product Owner - is responsible for working with the user group to determine what features will be in the product release. Some of the responsibilities:

- Develop the direction and strategy for the products and services, including the short and long-time goals;
- Provide or have access to knowledge about the product or the service;
- Understand and explain customer needs for the Development team;

Scrum Master - is the facilitator for an agile development team. Some of the responsibilities:

- Act as a coach, helping the team to follow scrum values and practices;
- Help to remove impediments and protect the team from external interferences;
- Promote a good cooperation between the team and stakeholders;

Scrum Team - is formed by 3 to 9 people who MUST fulfill all technical needs to deliver the product or the service. They will be guided directly by the Scrum Master, but they will not be directly managed. They must be self-organized, versatile, and responsible enough to complete all required tasks.

Artifacts

The SCRUM artifacts are used to help define the workload coming into the team and currently being worked upon the team.

The main artifacts:

- Product backlog - a collection of user stories which present functionalities required/wanted by the product team. Usually the product owner takes responsible for this list.

- Sprint backlog - a collection of stories which could be included in the current sprint.

User Stories

A User Story is a simple and quick description of a specific way that the user will use the software. Generally between one and four sentences long.

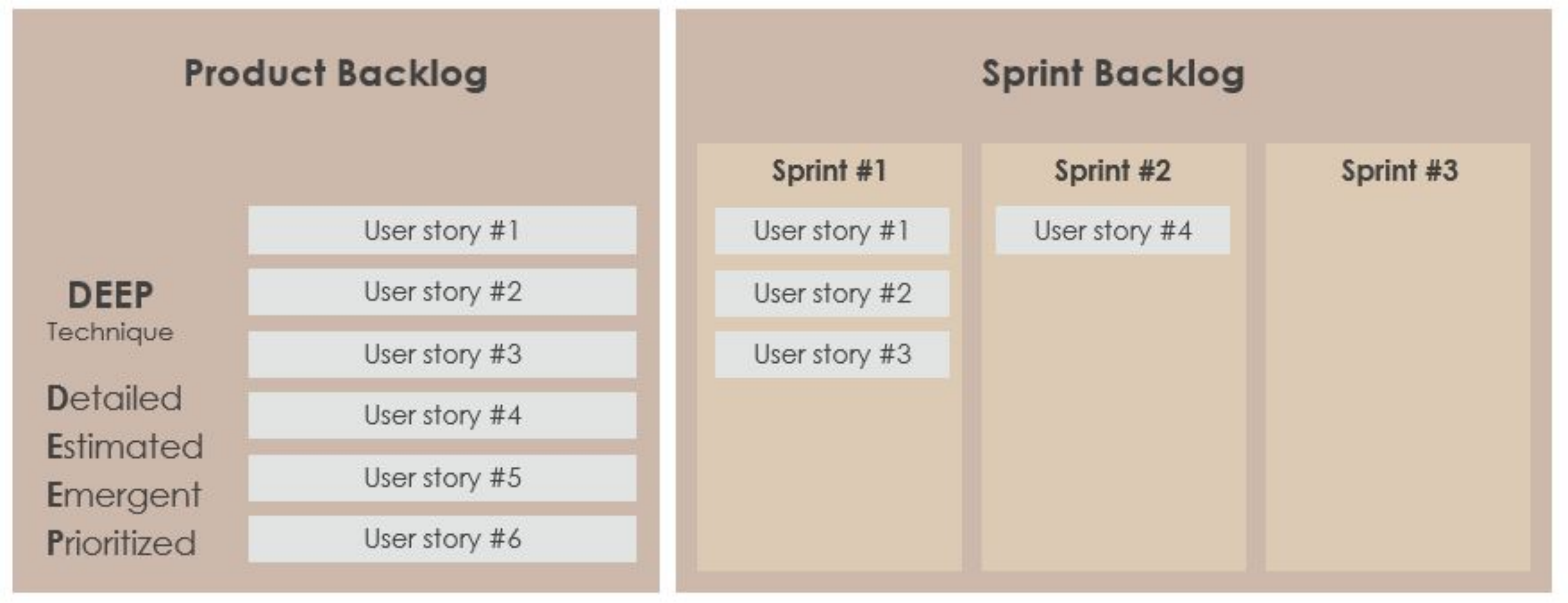
Can generally follow a template:

As a <*type of user*>,
I want to <*specific action I'm taking*>
so that <*what I want to happen as a result*>.

e.g. "As a customer, I want to be able to create an account so that I can see the purchases I made in the last year to help me budget for next year."

Assign a value to estimate the effort needed to elaborate a user story (e.g., 1 to 5).

Artifacts: Product Backlog and Sprint Backlog



Product Backlog

DEEP
Technique

Detailed
Estimated
Emergent
Prioritized

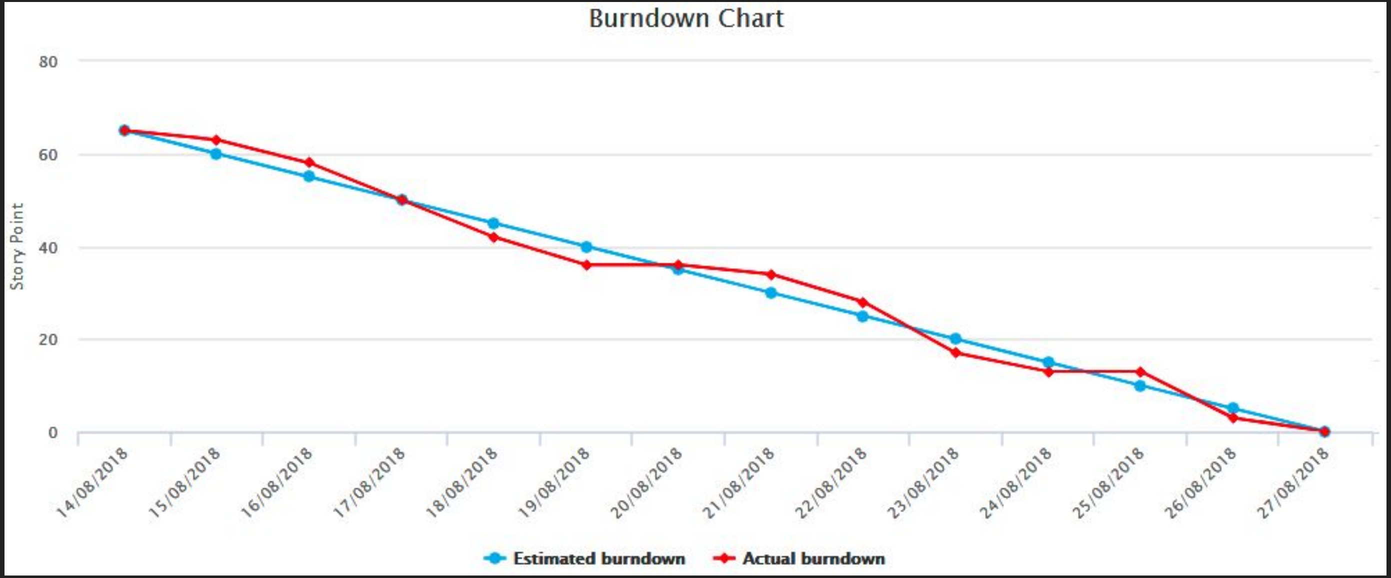
- User story #1
- User story #2
- User story #3
- User story #4
- User story #5
- User story #6

Sprint Backlog

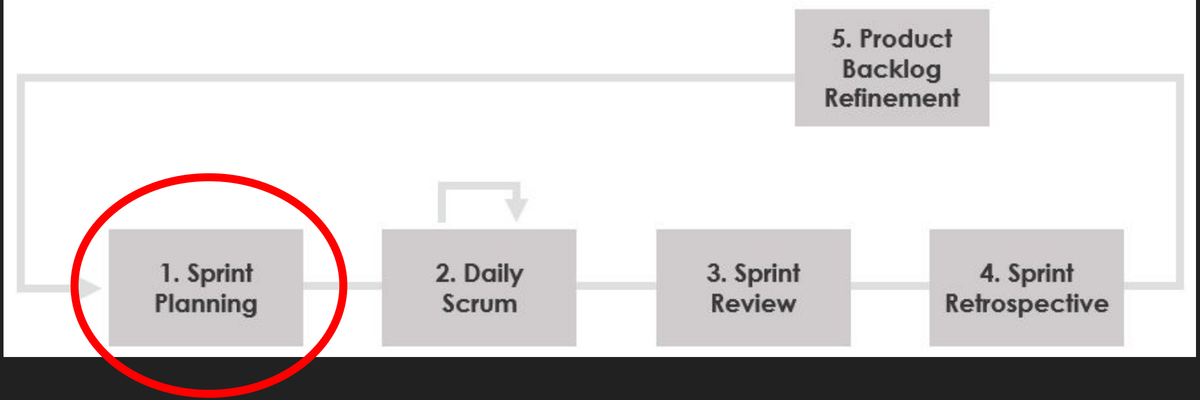
Sprint #1	Sprint #2	Sprint #3
User story #1	User story #4	
User story #2		
User story #3		

Artifacts: Burn-down chart

A burn-down chart is a graphical representation of work left to do versus time.
It is useful for predicting when all of the work will be completed.

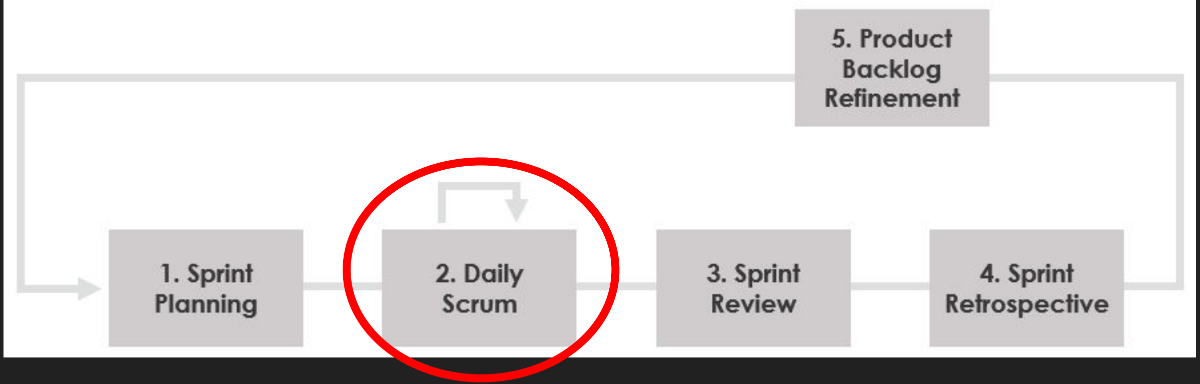


Events



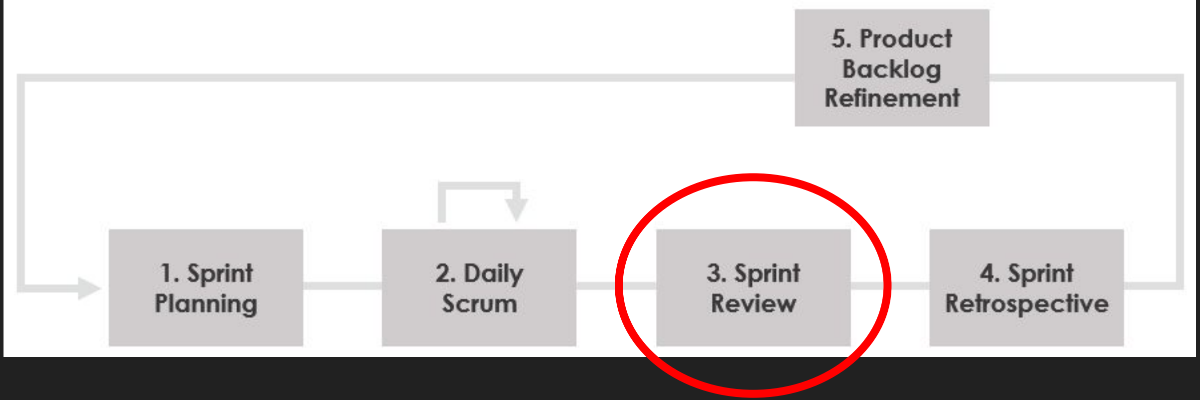
- All sprints begin with planning.
- The team needs to identify and commit to which items are going to be delivered as part of the sprint.
- Here the Scrum master has a main role

Events



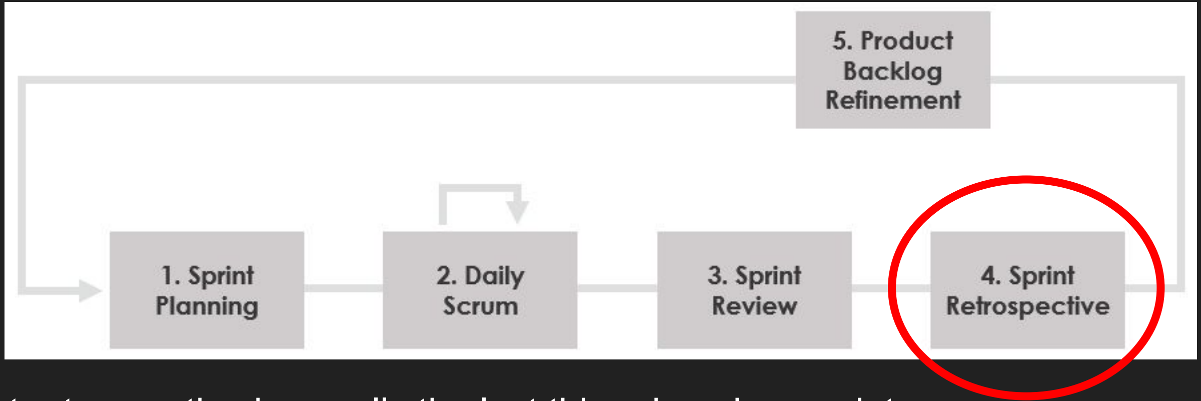
- The aim of this meeting is to ensure everyone within the team knows the status of the tasks accomplished (done) and of those in progress.
- The team has to answer the following questions:
  - What have we done until now?
  - What are we going to do today?
  - What are the impediments?
- No longer than 3 minutes per person
- The SCRUM master must where possible mitigate outside interruptions and distractions to the team

Events



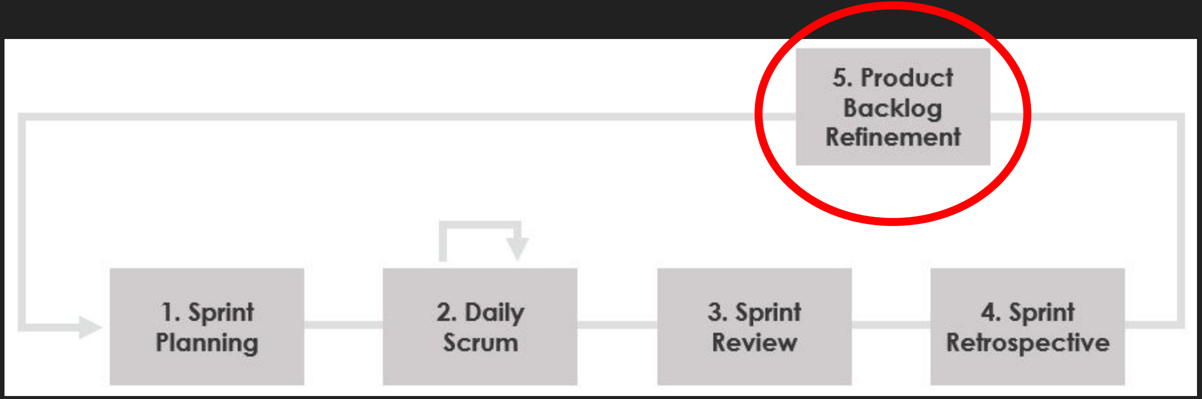
- A Sprint Review/Demo meeting is held at the end of the Sprint to inspect the Increment.
- The Team demonstrates the Increment with focus on the Sprint Goal according to the Definition of Done.
- The Product Owner reviews and accepts the delivered Increment.

Events



- The sprint retrospective is usually the last thing done in a sprint.
- You can schedule a scrum retrospective for up to an hour, which is usually quite sufficient.
- The retrospective gives the team the opportunity to identify 3 key aspects:
  - What should starting doing?
  - What did not go well (and stop doing again)?
  - What went well (and should keep doing)?
- Continually improve the team efficiency.

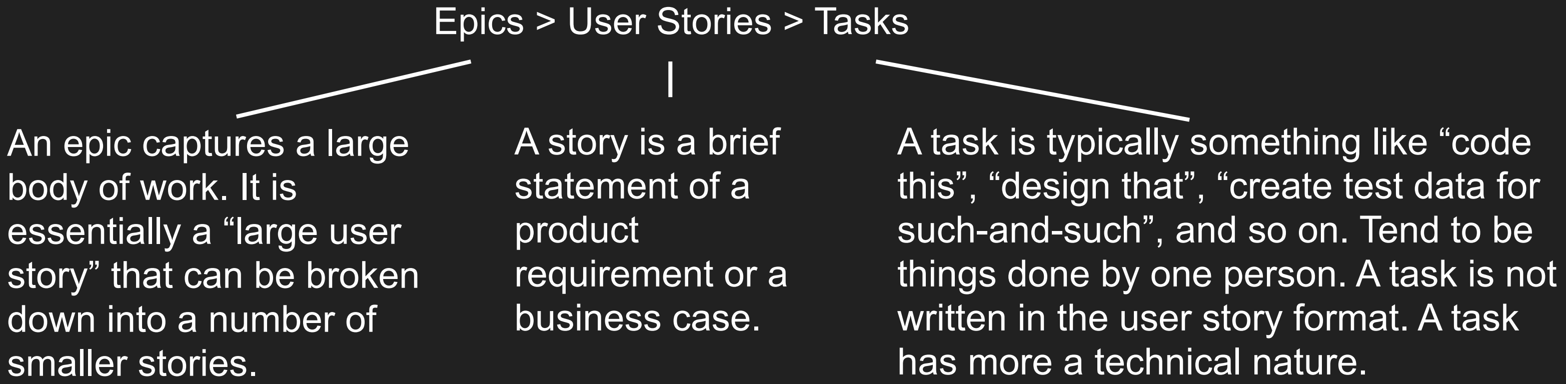
Events



- Think of the backlog as the roadmap of the project.
- As the team collaborates to create a list of everything that needs to be built or done for project completion, this list can be modified and added to throughout the project to ensure that all of the necessary needs of the project are met.

Sprint

In the Scrum Framework all activities needed for the implementation of entries from the Scrum Product Backlog are performed within Sprints (also called ‘Iterations’). Sprints are always short: normally about 2-4 weeks.



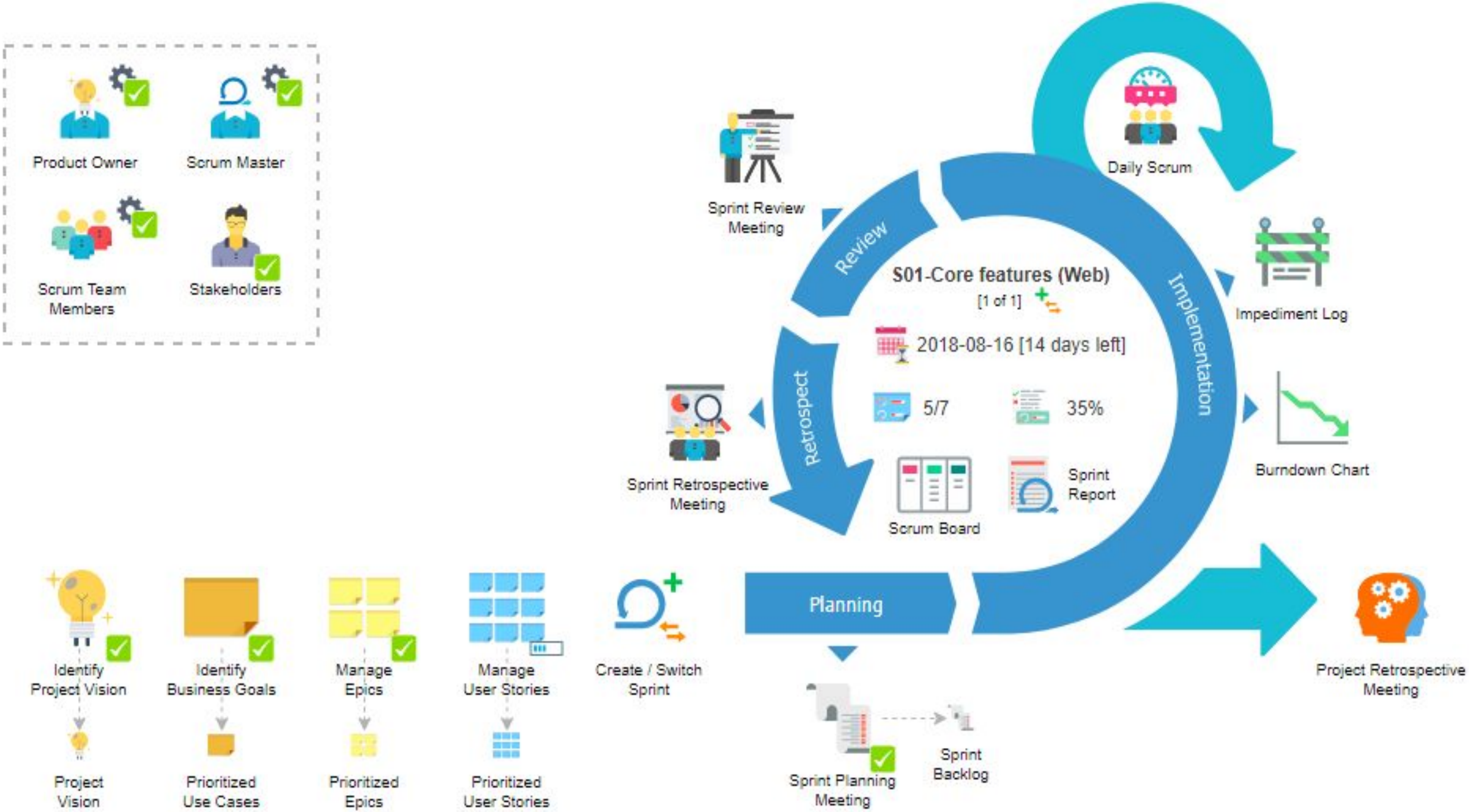
Epics > User Stories > Tasks

An epic captures a large body of work. It is essentially a “large user story” that can be broken down into a number of smaller stories.

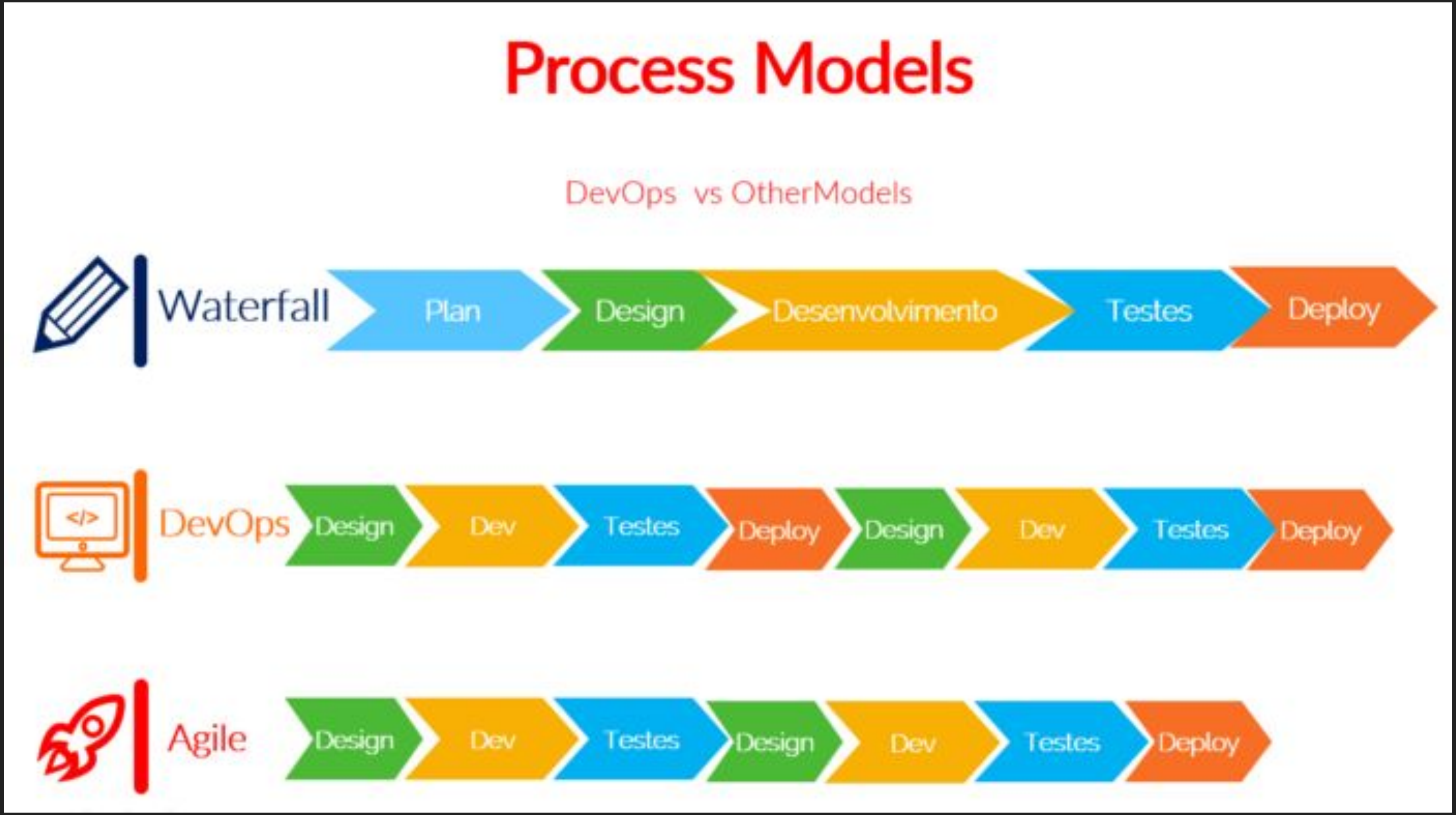
A story is a brief statement of a product requirement or a business case.

A task is typically something like “code this”, “design that”, “create test data for such-and-such”, and so on. Tend to be things done by one person. A task is not written in the user story format. A task has more a technical nature.

Experienced Scrum Teams spend time and effort to break down complex and larger items (i.e user features or epics) into smaller user stories (or subsequently breaking down into tasks, or subtasks).


Product Owner
Scrum Master
Scrum Team Members
Stakeholders
Sprint Review Meeting
Daily Scrum
Review
Implementation
S01-Core features (Web)
[1 of 1]
Impediment Log
2018-08-16 [14 days left]
Retrospect
5/7
35%
Burndown Chart
Sprint Retrospective Meeting
Scrum Board
Sprint Report
Identify Project Vision
Identify Business Goals
Manage Epics
Manage User Stories
Create / Switch Sprint
Planning
Project Retrospective Meeting
Project Vision
Prioritized Use Cases
Prioritized Epics
Prioritized User Stories
Sprint Planning Meeting
Sprint Backlog

Recap

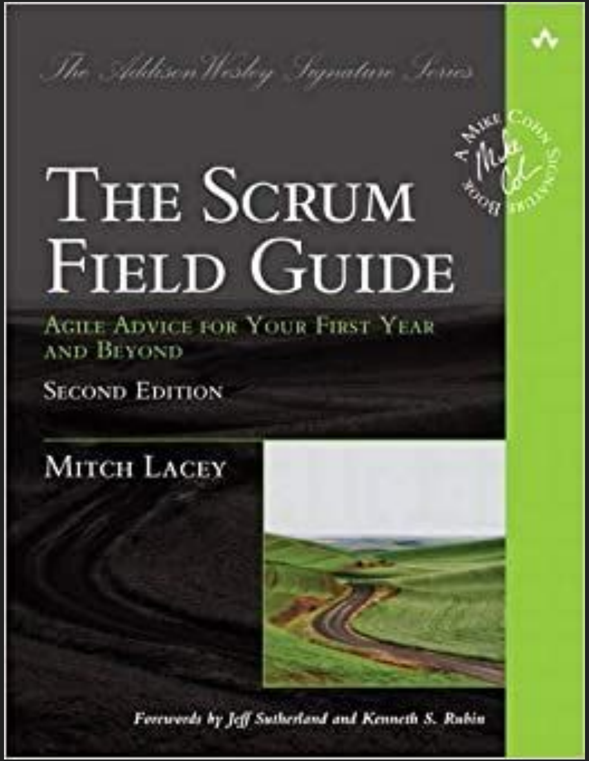

Process Models
DevOps vs OtherModels
Waterfall
Plan
Design
Desenvolvimento
Testes
Deploy
</>
DevOps
Design
Dev
Testes
Deploy
Design
Dev
Testes
Deploy
Agile
Design
Dev
Testes
Design
Dev
Testes
Deploy

Additional Materials

Scrum Field Guide,

The: Agile Advice for Your First Year and Beyond (Addison-Wesley Signature Series (Cohn)) 2nd Edition

By Mitch Lacey

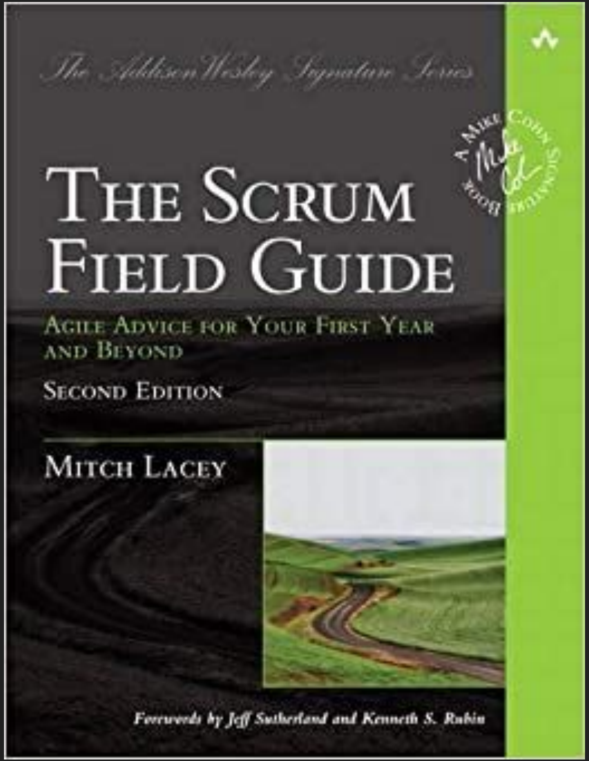

The Addison Wesley Signature Series
A Mike Cohn Signature Book
The Scrum Field Guide
Agile Advice for Your First Year and Beyond
Second Edition
Mitch Lacey
Forewords by Jeff Sutherland and Kenneth S. Rubin

Additional Materials

Scrum Field Guide,

The: Agile Advice for Your First Year and Beyond (Addison-Wesley Signature Series (Cohn)) 2nd Edition

By Mitch Lacey


The Addison Wesley Signature Series
A Mike Cohn Signature Book
The Scrum Field Guide
Agile Advice for Your First Year and Beyond
Second Edition
Mitch Lacey
Forewords by Jeff Sutherland and Kenneth S. Rubin

https://www.mountaingoatsoftware.com/

By Mike Cohn

Exam

The exam involves the development of a software project by following the methodologies introduced during lecture hours with the objective of supporting a “Smart Parking” system.

Exam Evaluation:

- Git Usage
- Testing
- SCRUM application/Sprint management
- Presentation
- Overall Project

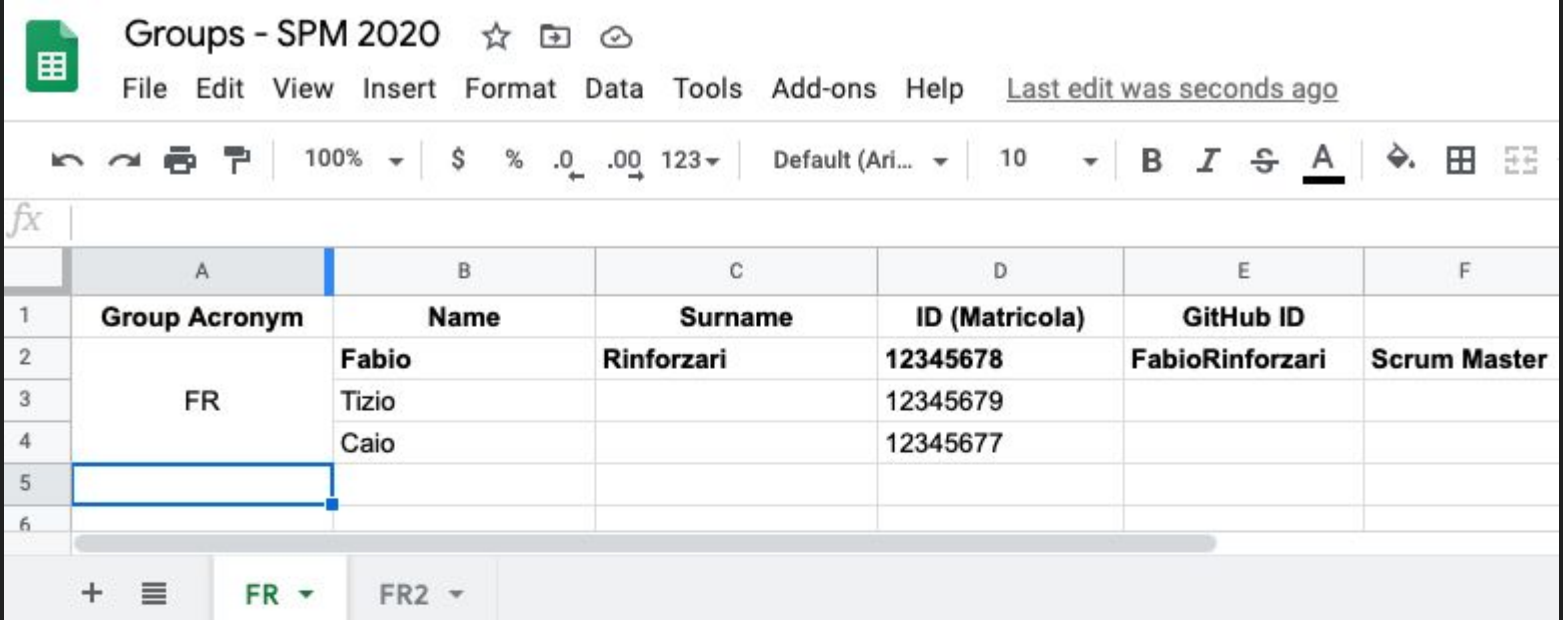
Group Formation

- Group of 2 or 3 students have to be formed
- A Project will be assigned
- A GitHub account is needed for every student
- A GitHub repository will be assigned to each group

Google Sheets

https://docs.google.com/spreadsheets/d/1tOdMoBKzBjcHGQI3ACmrkkcaFbc2TuPfgmFQM2jL9BU

https://tinyurl.com/yynuktcl



Group Acronym	Name	Surname	ID (Matricola)	GitHub ID	
FR	**Fabio**	**Rinforzari**	**12345678**	**FabioRinforzari**	**Scrum Master**
	Tizio		12345679		
	Caio		12345677		

Any Question?