CSE- 321

Software Engineering

Lecture: 08

Software Processes
(part-02)

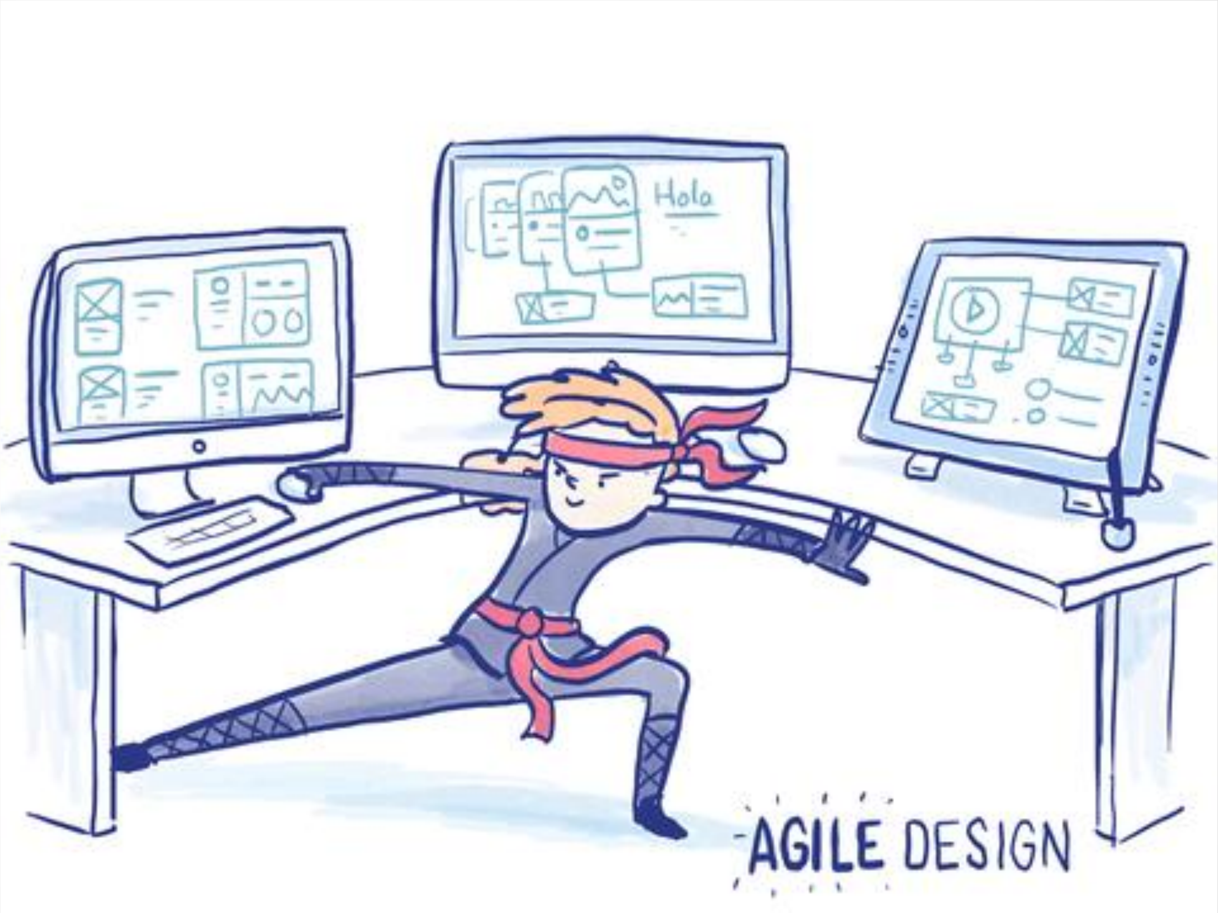
Fahad Ahmed

Lecturer, Dept. of CSE

E-mail: fahadahmed@uap-bd.edu

Lecture Outlines

❖**Agile Model**



Rather than doing all of one thing at a time, is better to do a little of everything all the time. :)

Agile Model

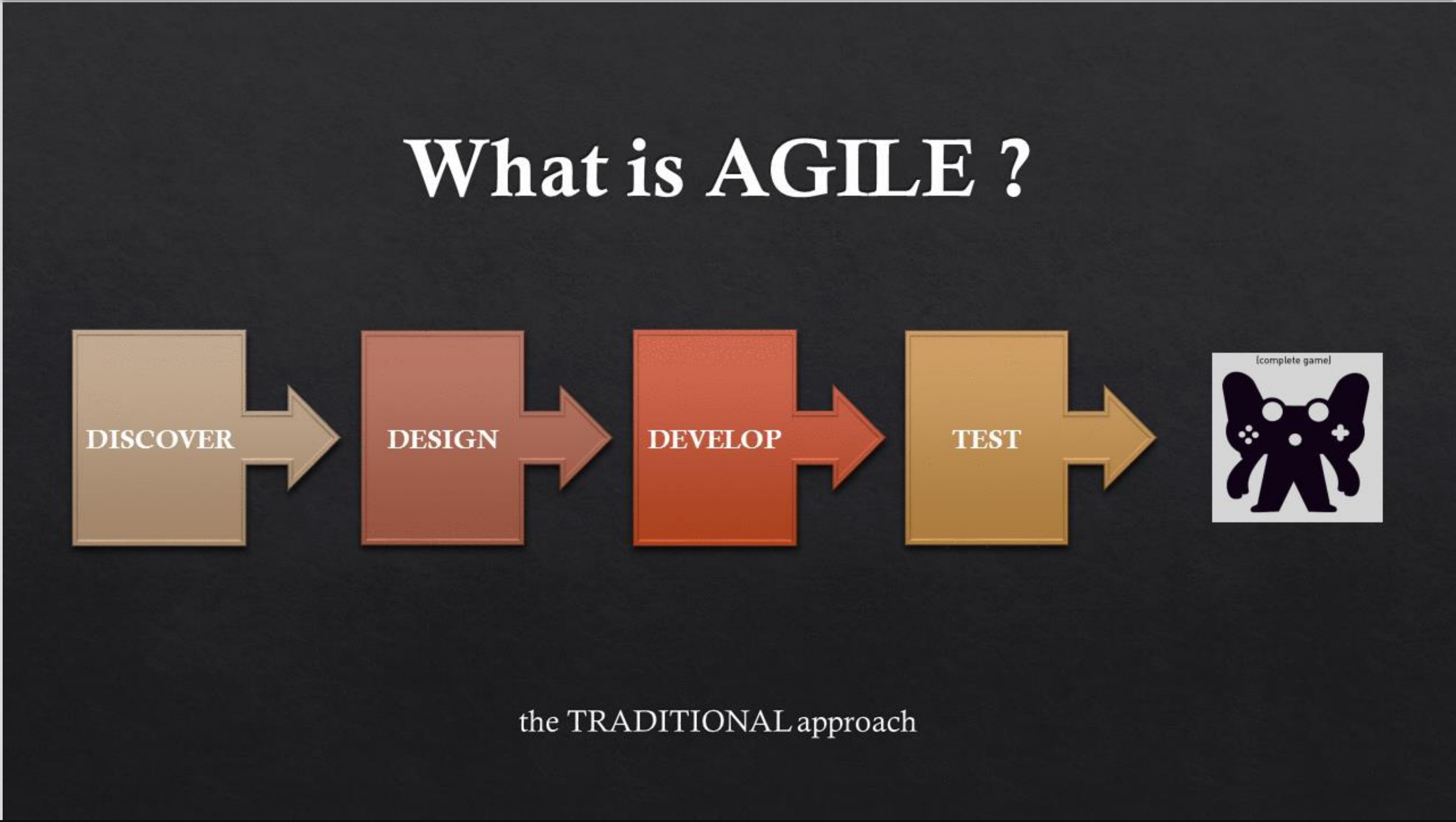
Agile model is a combination of **iterative** and **incremental** process models with focus on process adaptability and customer satisfaction by **rapid delivery** of working software product.

This is where the agile software development comes to the **rescue**. It was specially designed to curate the needs of the **rapidly changing environment** by embracing the idea of incremental development and develop the actual final product.

Agile Methods break the product into **small incremental builds**. These builds are provided in iterations.

At the end of the iteration a working product is **displayed to the customer** and **important stakeholders.**

Agile Model



What is AGILE ?

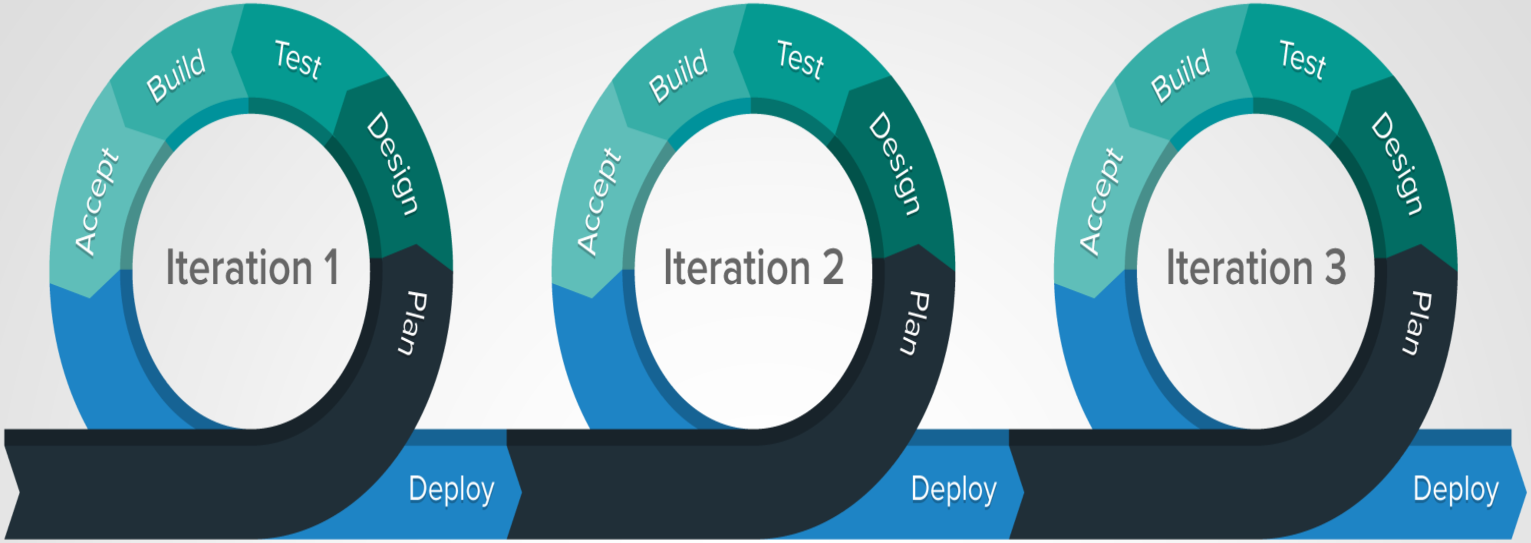
DISCOVER → DESIGN → DEVELOP → TEST →

(complete game)

the TRADITIONAL approach

Rather than doing all of one thing at a time, is better to do a little of everything all the time.

Agile Model



Agile Process : Development + Testing

- In Agile, a company releases the application **with some high priority features** in the first iteration.
- After its release, the **end-users** or the customers **give you feedback** about the performance of the application.
- Then you **make the necessary changes** into the application along with some new features and the application is again released which is the second iteration.
- You repeat this entire procedure until you achieve the desired software quality.

Agile Model

Agile model believes that every project needs to be **handled differently** and the existing methods need to be tailored to best suit the project requirements.

In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Iterative approach is taken and working software build is **delivered after each iteration**. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Agile Model

Agile Manifesto

The Agile Manifesto’s core statement says: – “***We are uncovering better ways of developing software by doing it and helping others do it. Through this work, we have come to value***:”

The agile software development emphasizes on four core values.

1. **Individuals and interactions** over processes and tools.
2. **Working software** over comprehensive documentation.
3. **Customer collaboration** over contract negotiation.
4. **Responding to change** over following a plan.

https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm

Agile Model : Case Study

A Software company named **ABC** wants to make a new web browser for the latest release of its operating system. The deadline for the task is 10 months. The company's head assigned two teams named **Team A** and **Team B** for this task.

In order to motivate the teams, the company head says that the first team to develop the browser would be given a salary hike and a one week full sponsored travel plan.

The team A decided to play by the book and decided to choose the **Waterfall model** for the development. Team B after a heavy discussion decided to take a leap of faith and choose **Agile as their development model**.

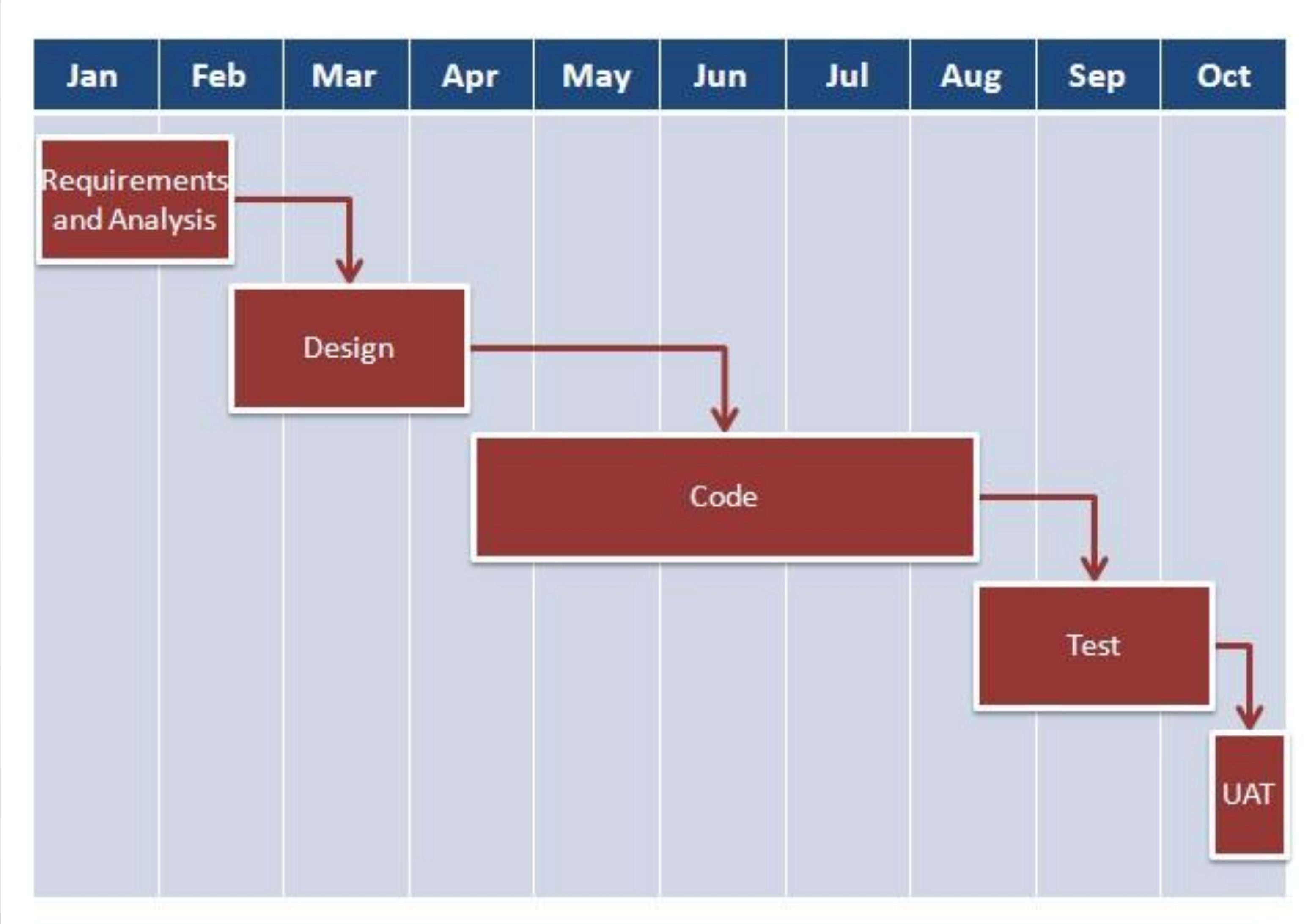
Agile Model : Case Study

In traditional Waterfall model –

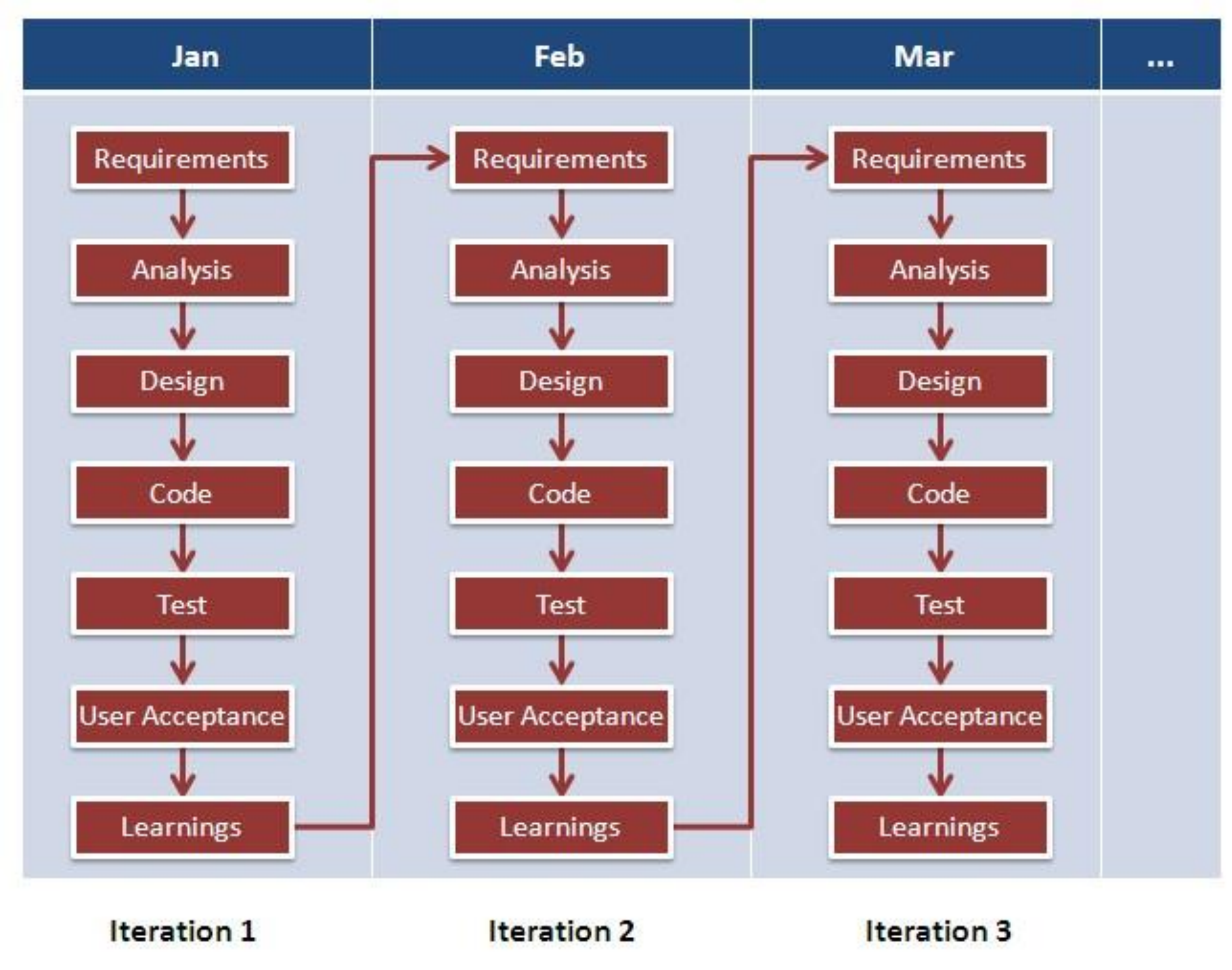
- At a high level, the project teams would spend 15% of their time on gathering requirements and analysis (1.5 months)
- 20% of their time on design (2 months)
- 40% on coding (4 months) and unit testing
- 20% on System and Integration testing (2 months).
- At the end of this cycle, the project may also have 2 weeks of User Acceptance testing by marketing teams.
- In this approach, the customer does not get to see the end product until the end of the project, when it becomes too late to make significant changes.

Agile Model : Case Study

The image below shows how these activities align with the project schedule in traditional software development.

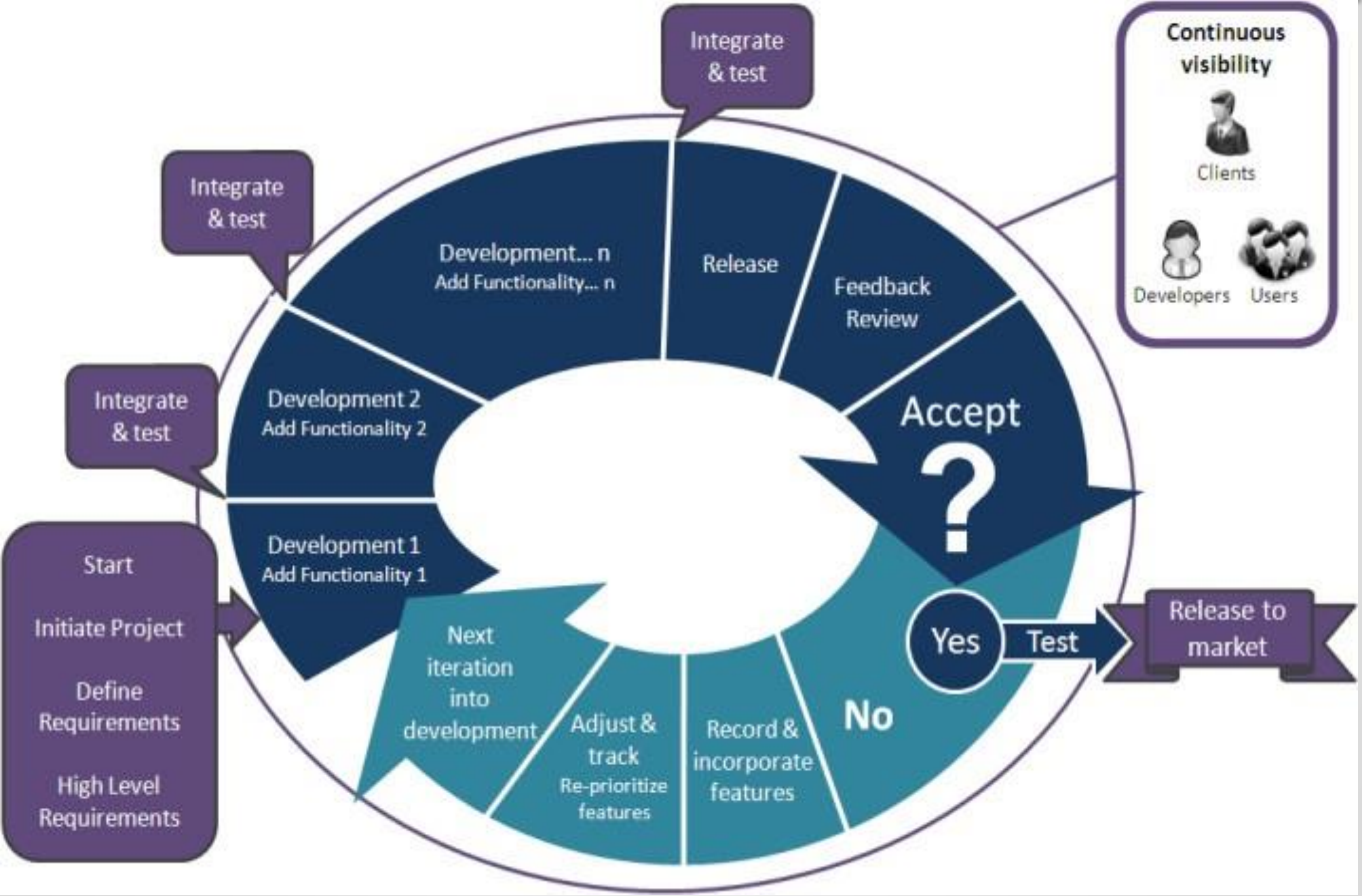


Agile Model : Case Study



Jan	Feb	Mar	...
Requirements	Requirements	Requirements	
Analysis	Analysis	Analysis	
Design	Design	Design	
Code	Code	Code	
Test	Test	Test	
User Acceptance	User Acceptance	User Acceptance	
Learnings	Learnings	Learnings	
Iteration 1	Iteration 2	Iteration 3	

Agile Model : Case Study



Agile Model : Case Study

With **Agile development** methodology –

- In the **Agile methodology**, each project is broken up into several ‘Iterations’.

- All Iterations should be of the same time duration (between 2 to 8 weeks).
- At the end of each iteration, a working product should be delivered.

- In simple terms, in the Agile approach the project will be broken up into 10 releases (assuming each iteration is set to last 4 weeks).

- Rather than spending 1.5 months on requirements gathering, in Agile software development, the team will decide the basic core features that are required in the product and decide which of these features can be developed in the first iteration.

- Any remaining features that cannot be delivered in the first iteration will be taken up in the next iteration or subsequent iterations, based on priority.

- At the end of the first iterations, the team will deliver a working software with the features that were finalized for that iteration.

Agile Model : Case Study

- There will be 10 iterations and at the end of each iteration the customer is delivered a working software that is incrementally enhanced and updated with the features that were shortlisted for that iteration

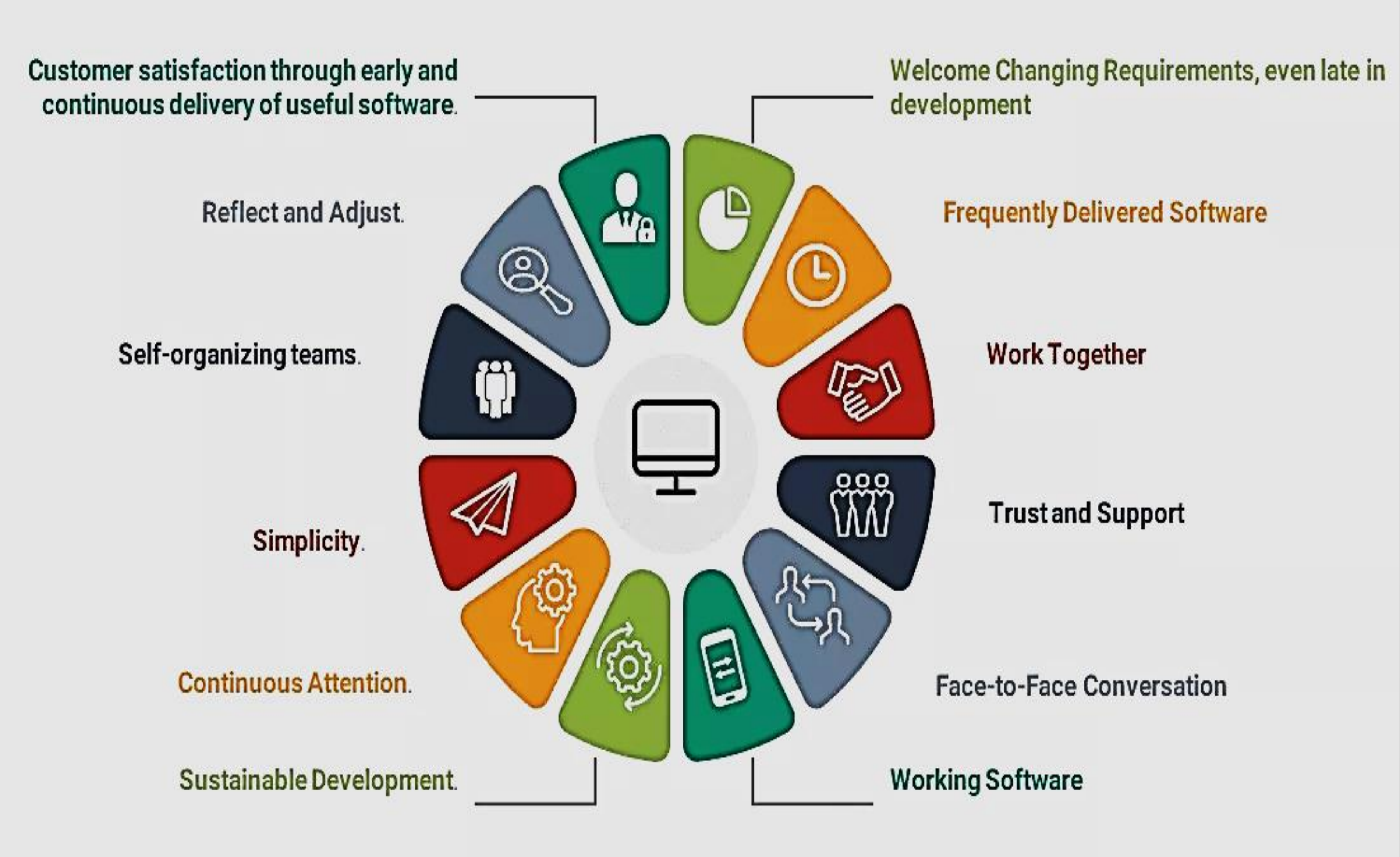
- This approach allows the customer to interact and work with functioning software at the end of each iteration and provide feedback on it. This approach allows teams to take up changes more easily and make course corrections if needed. In the Agile approach, software is developed and released incrementally in the iterations.

- **Agile development has become common place in IT industry.**
- **In a recent survey over 52% of respondents said that their company practiced Agile development in one form or another.**

Agile Model

Principles of Agile model:

https://www.toolsqa.com/agile/agile-methodology/



Customer satisfaction through early and continuous delivery of useful software.

Welcome Changing Requirements, even late in development

Frequently Delivered Software

Work Together

Trust and Support

Face-to-Face Conversation

Working Software

Sustainable Development.

Continuous Attention.

Simplicity.

Self-organizing teams.

Reflect and Adjust.

Agile Model

Principles of Agile model:

1. Our highest priority is to **satisfy the customer** through early and continuous delivery of valuable software.

2. **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.

3. **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4. Business people and developers must work together daily throughout the project.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most efficient and effective method of conveying information to and within a development team is **face–to–face conversation**.

Agile Model

Phases of Agile Model:

Following are the phases in the Agile model are as follows:

- Requirements gathering
- Design the requirements
- Construction/ iteration
- Testing/ Quality assurance
- Deployment
- Feedback

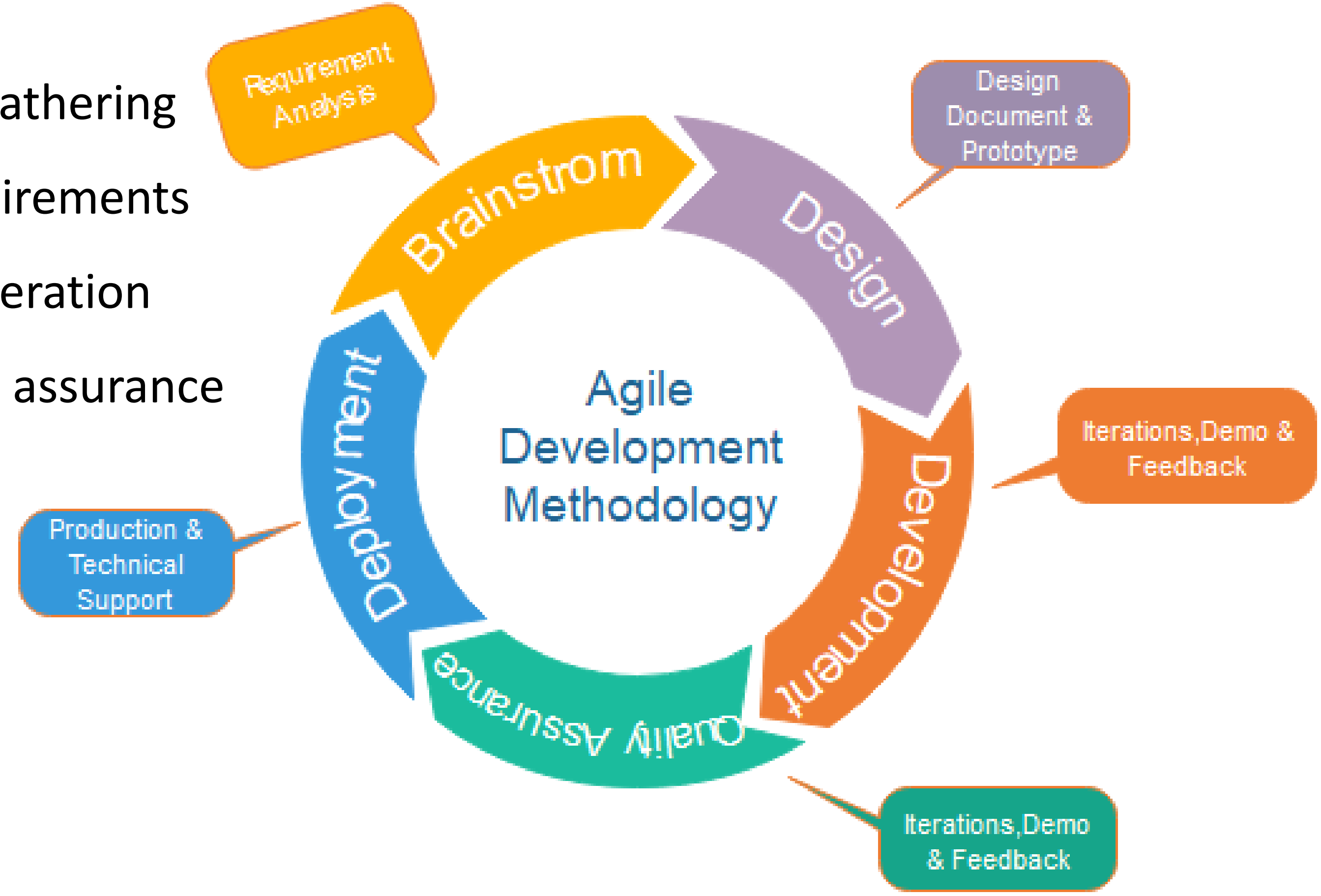

Requirement Analysis
Design Document & Prototype
Brainstrom
Design
Agile Development Methodology
Iterations,Demo & Feedback
Development
Production & Technical Support
Deployment
Quality Assurance
Iterations,Demo & Feedback

Fig. Agile Model

Agile Model

What is a real-time example of software which is developed using Agile methodology?

- The UK Government (HMRC) tax platform
- Network International’s payment gateway
- Netflix systems
- XBox Live was built using one week sprints in Scrum.
- www.study.com is an example. The company deploys to their site on average five times a day, users a mix of Scrum and XP

Agile Model

Agile Methodologies(Framework):

- **Scrum**
- **eXtreme Programming(XP)**
- DSDM (Dynamic Software Development Method)
- FDD (Feature-Driven Development)
- ASD (Adaptive System Development)
- Crystal Methodology
- Kanban

https://www.javatpoint.com/software-engineering-agile-model

Agile Model

What is Scrum?

Scrum is one of the **most commonly used Agile methodologies**.

Scrum focuses on Team

Scrum is a framework that helps teams work together.

It is a method that concentrates specifically on how to **manage tasks within a team-based development** environment.

Agile Model

What is Scrum?

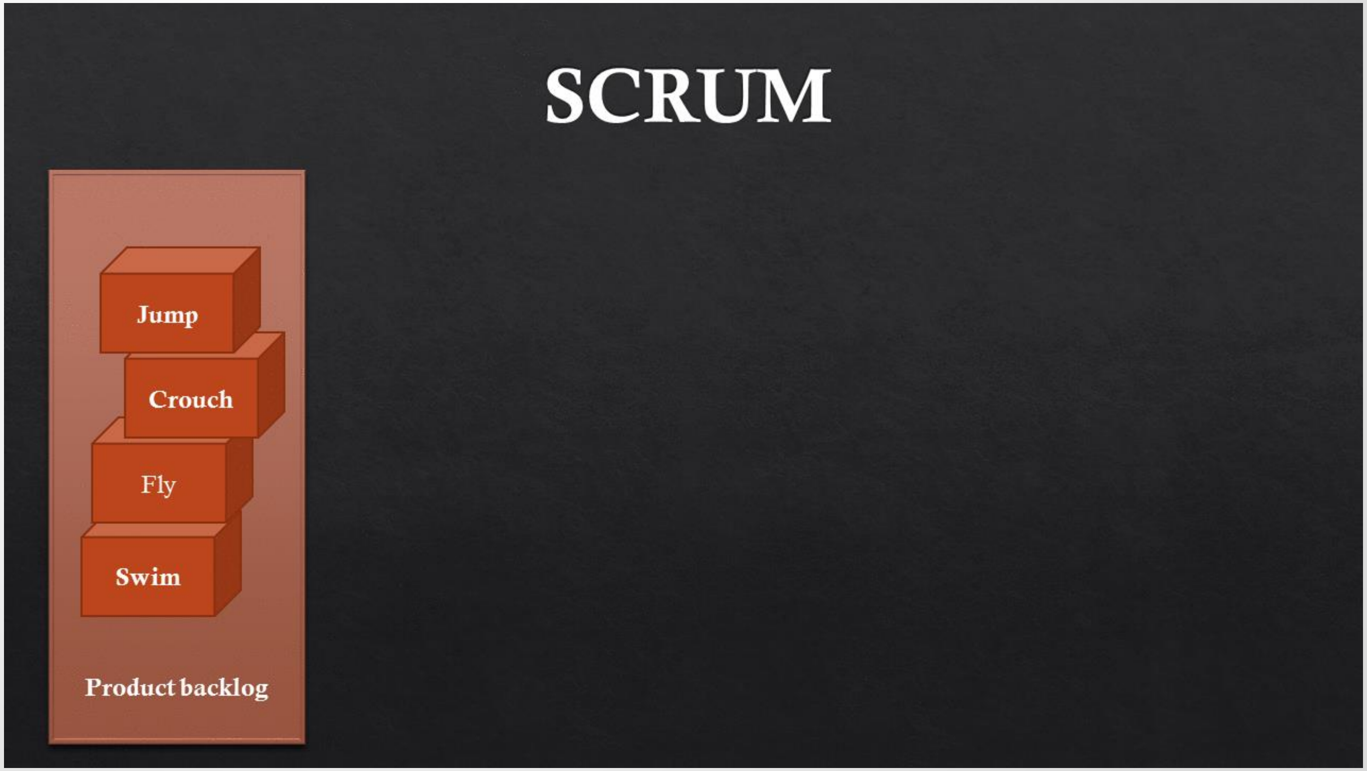
SCRUM is an iterative and incremental Agile process that produces demonstrable working software every 2-4 weeks. Being results- and commitment-oriented it has a long history of success on a wide variety of projects, including game development.

Agile Model: Scrum

- Each iteration of a scrum is known as **Sprint**
- Scrum has three roles product owner, Scrum Master and The Team Members.

- Scrum uses three artefacts Product backlog, Sprint Backlogs and Burn down Chart to guide the team during the sprint.
- Scrum defines three ceremonies Sprint Planning, Daily Scrum and Sprint Review & Retrospective.

- **Product backlog** is a list where all details are entered to get the end-product



case: games(output)

Agile Model: Scrum

- The product owner takes care of the Product Backlogs. The Backlog is the **list** of the every desirable outcome user expect from the product. This is the centrally TO DO LIST of GOAL.
- During each Sprint, top user stories of Product backlog are selected and turned into **Sprint backlog**
- Team works on the defined **sprint backlog**
- Team checks for the daily work
- At the end of the sprint, team delivers product functionality

Agile Model: Scrum

Product Owner

- Is ***responsible for the profitability*** of the product (ROI)
- ***Prioritizes features*** according to ***market value***
- Can ***change features*** and priority every sprint
- Accepts or rejects work results
- ***Defines the features of the product***, decides on release date and content

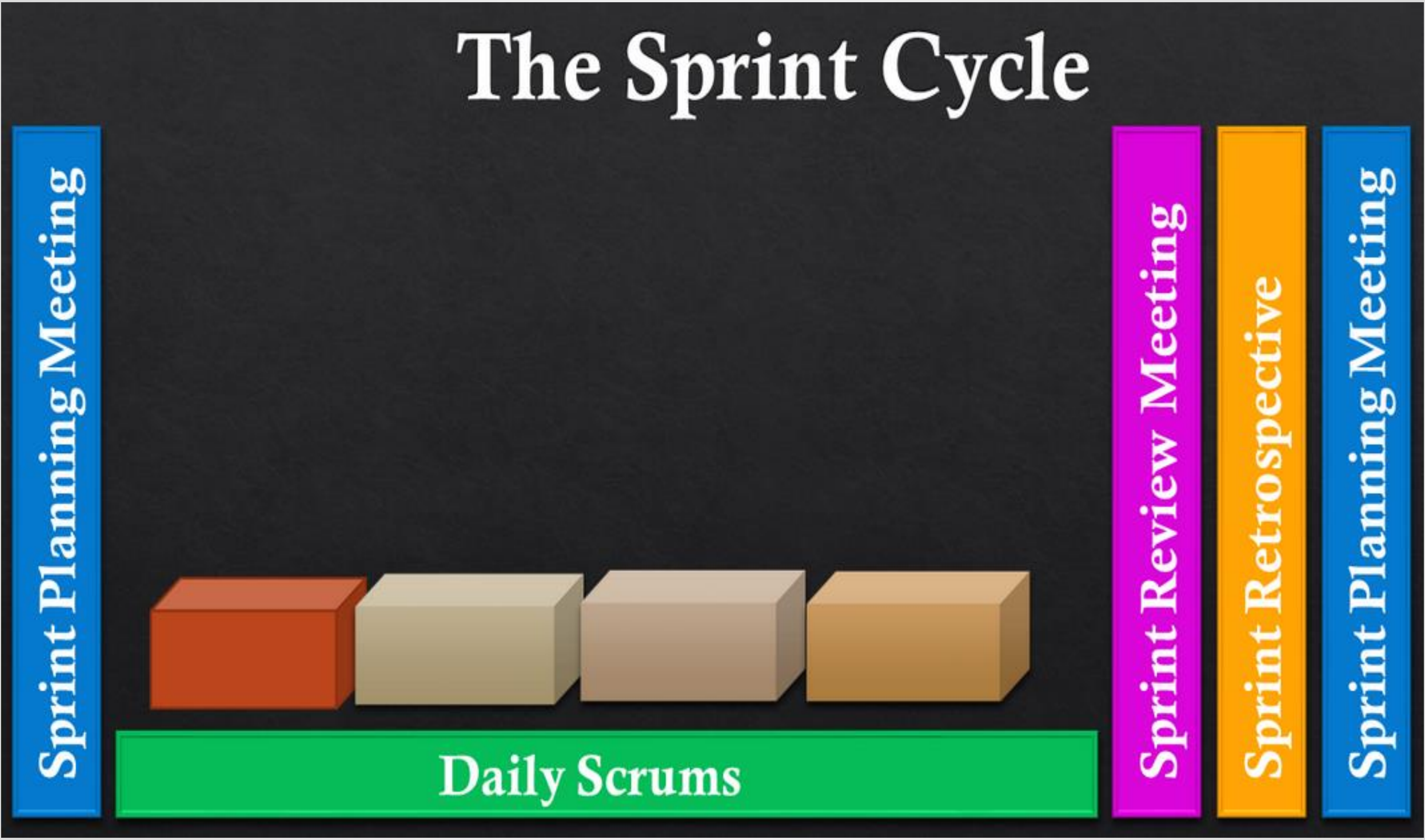
Scrum Master

- ***Ensures*** that the team is fully **functional** and **productive**
- ***Shields the team*** from external interferences
- Ensures that the process is followed.
- Enables close cooperation across all roles and functions and removes barriers
- ***Participates*** in daily scrum, sprint review and planning meetings

The Development Team

- ***Cross-functional***, seven plus/minus two ideally full-time members
- Selects the ***sprint backlog***
- ***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

Agile Model: Scrum



Agile Model: Scrum

Sprint Planning Meeting

- Product owner ***describes priorities***
- Team breaks product backlog items (features) into ***tasks***
- ***Team commits*** to some amount of work

Daily Scrum

- Three questions:
  - ***What did you do yesterday?***
  - ***What will you do today?***
  - ***What's in your way?***
- A commitment between peers, not a management status meeting

Agile Model: Scrum

Sprint Review Meeting

•Team demonstrates what was accomplished

•2-hour prep time rule

•No PowerPoint!

•Anyone can attend

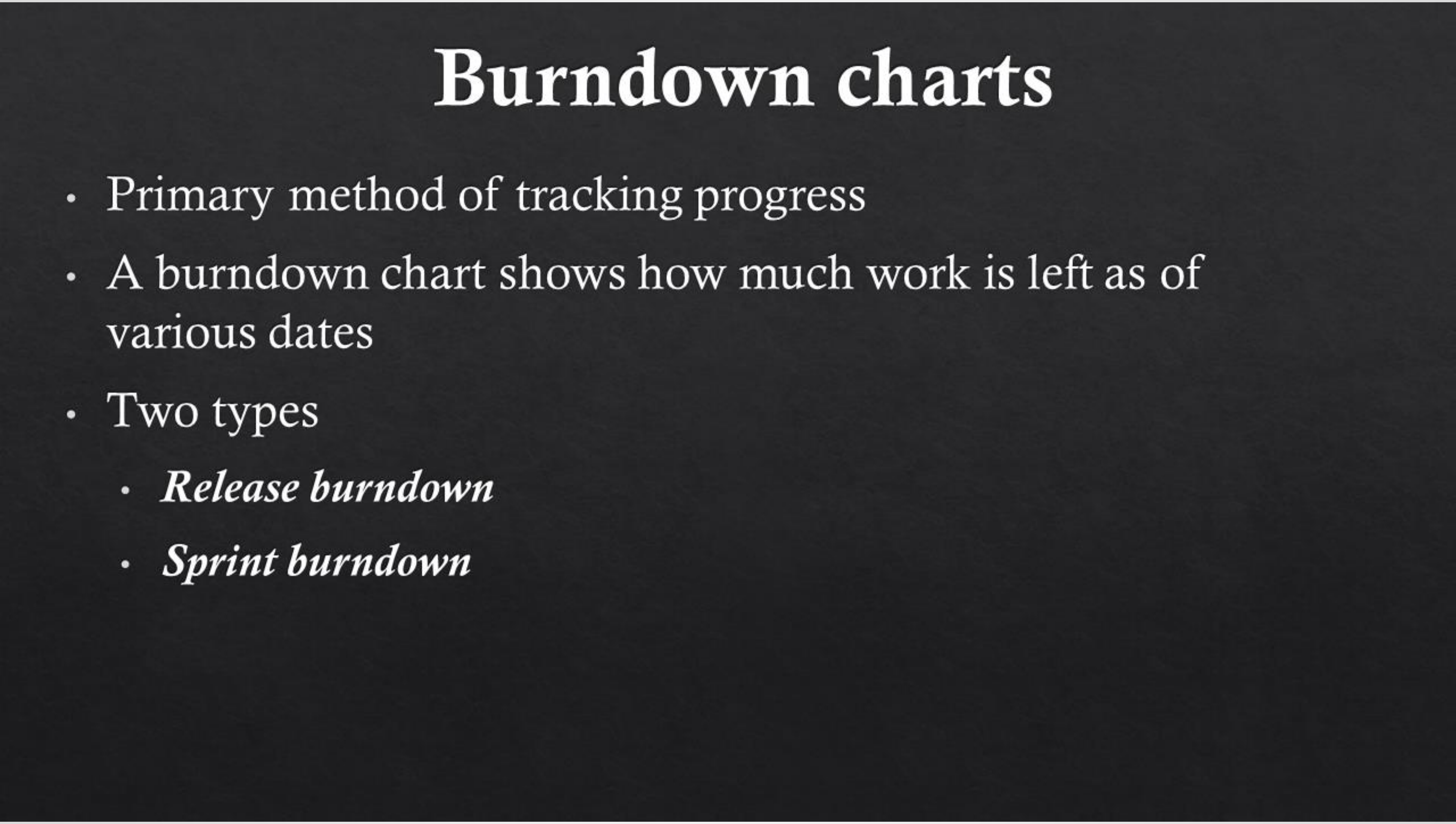
Sprint Retrospective

•Focus on continuous improvement

•Look back at the last sprint and ask what would we like to:

- ***What went well***
- ***What we need to improve & actions***
- ***Impediments***

Agile Model: Scrum

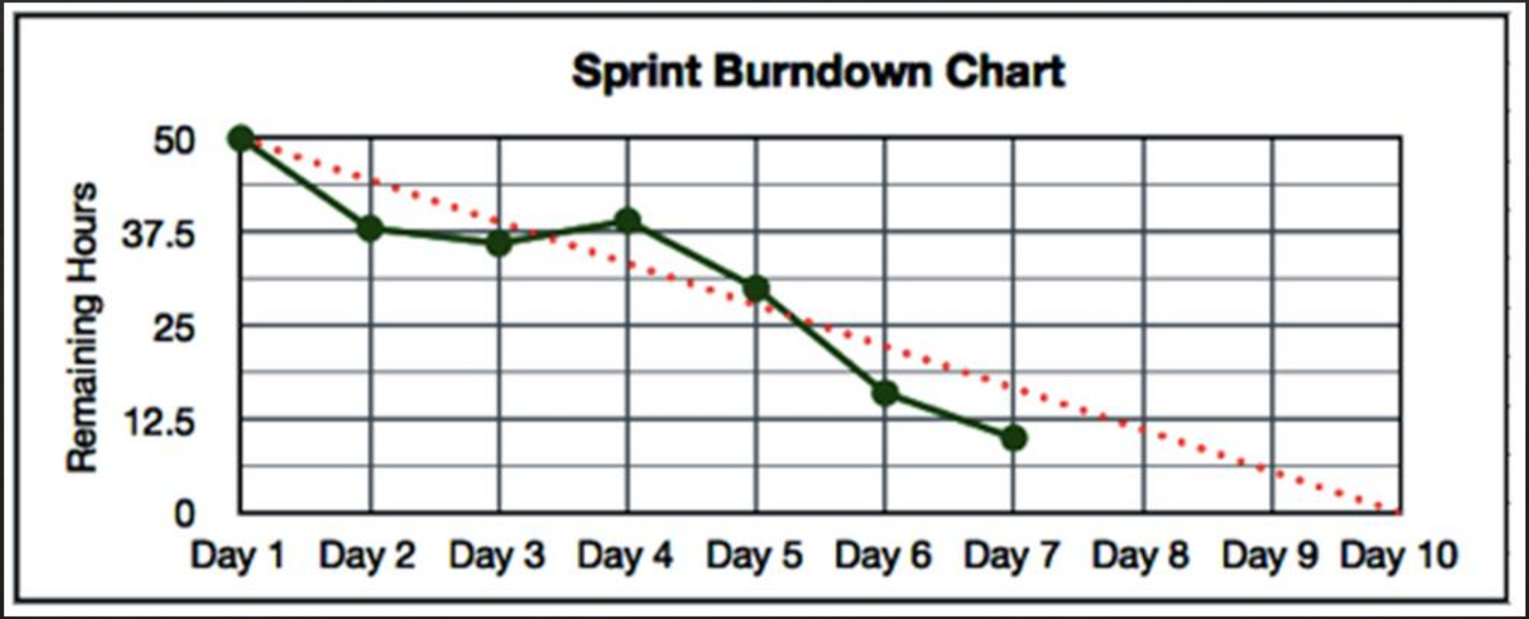


Burndown charts

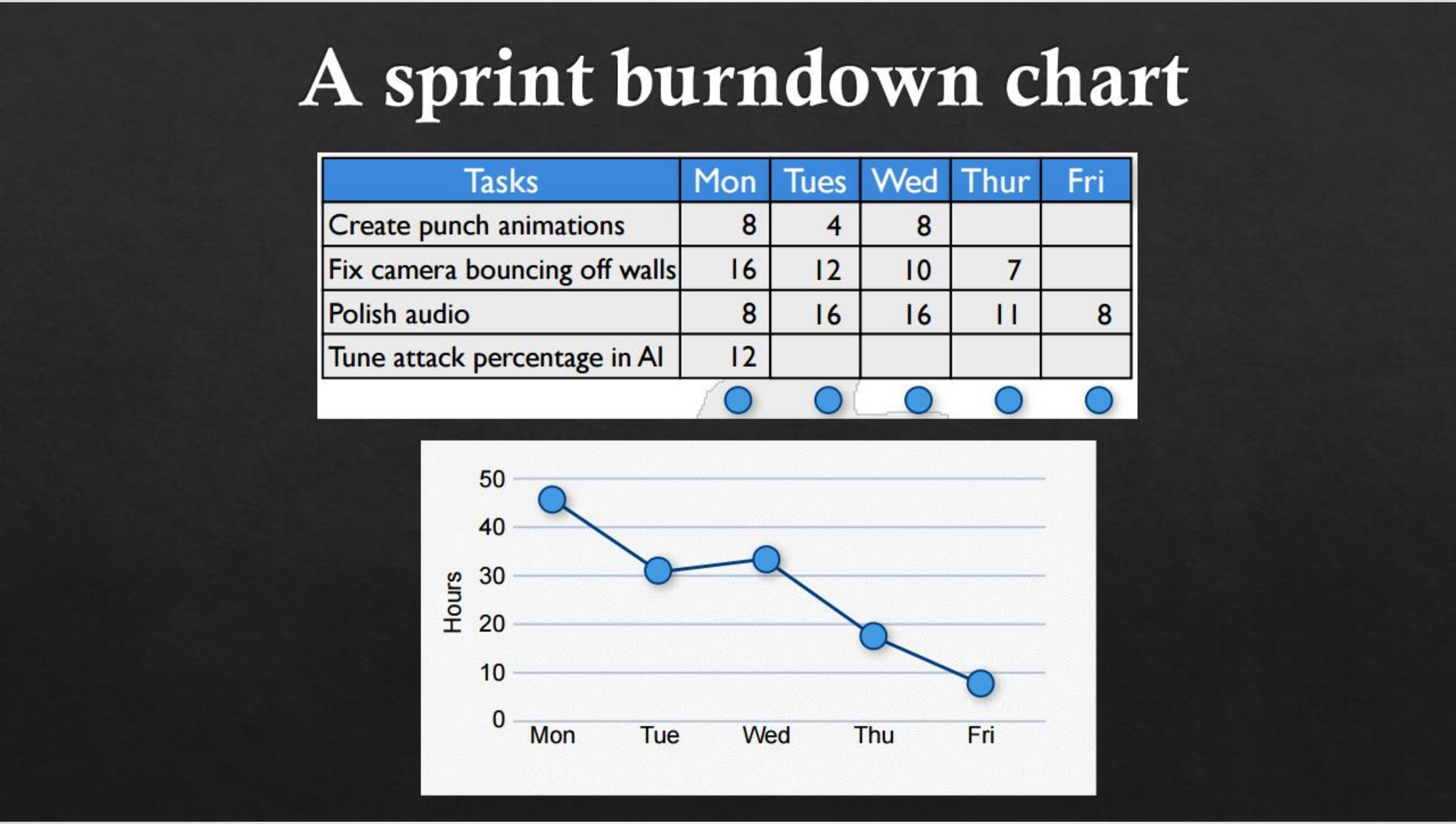
- Primary method of tracking progress
- A burndown chart shows how much work is left as of various dates
- Two types
  - *Release burndown*
  - *Sprint burndown*

Agile Model: Scrum

A sprint burndown chart



Agile Model: Scrum



A sprint burndown chart

Tasks	Mon	Tues	Wed	Thur	Fri
Create punch animations	8	4	8		
Fix camera bouncing off walls	16	12	10	7	
Polish audio	8	16	16	11	8
Tune attack percentage in AI	12				

Question:

1. How do I develop an Agile methodology mind-set while implementing a project?
2. Is Agile the best methodology for Mobile app development? Is Agile methodology appropriate for all projects?
3. Will agile methodology work for data warehousing projects?
4. Is it possible to run scrum without a scrum master? If yes, in what scenarios?
5. Does Scrum work for single person projects?
6. What kind of projects can benefit from XP? What projects are "too big" and therefore outside the scope of XP?
7. What is the next generation software development methodology after Agile?

https://www.quora.com/

Thanks to All