PMG 4101: Project Management

Content 2: Software Engineering Methodologies

Dr. Suman Ahmmed

Associate Professor, CSE, UIU

Director, Centre for Development of IT Professionals (CDIP)

Room: 1035, Email: suman@cse.uiu.ac.bd, Mobile/WhatsApp: 01765 049901

United International University | Dhaka, Bangladesh

Today's Agenda

01 Software Engineering Methodology

02 Agile Methodology

03 Waterfall Methodology

04 Rational Unified Process (RUP) Methodology

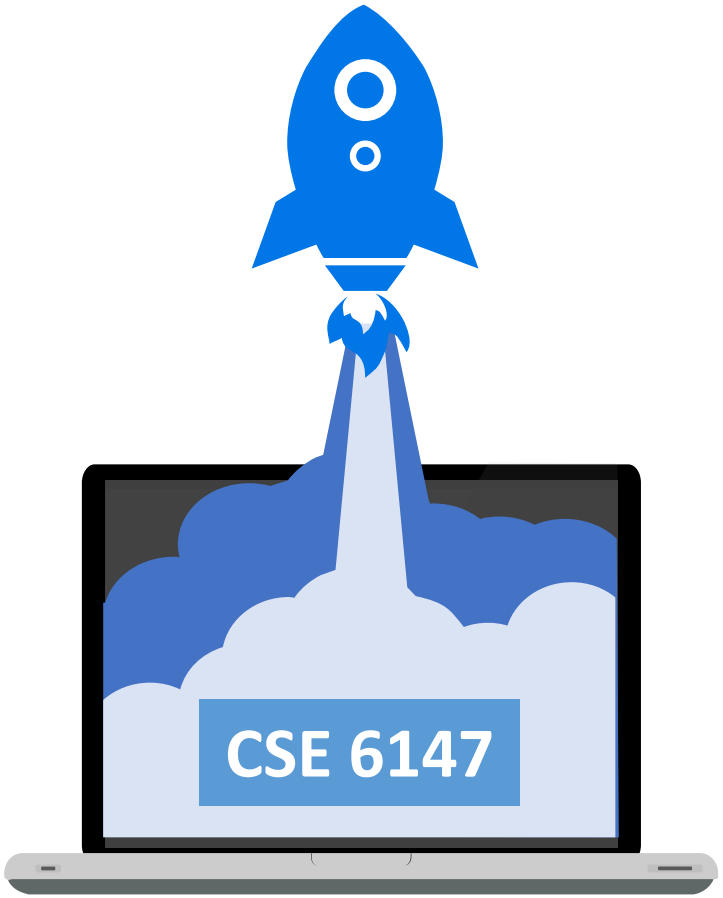
05 Prototype Model

06 Scrum Methodology

07 eXtreme Programming Methodology

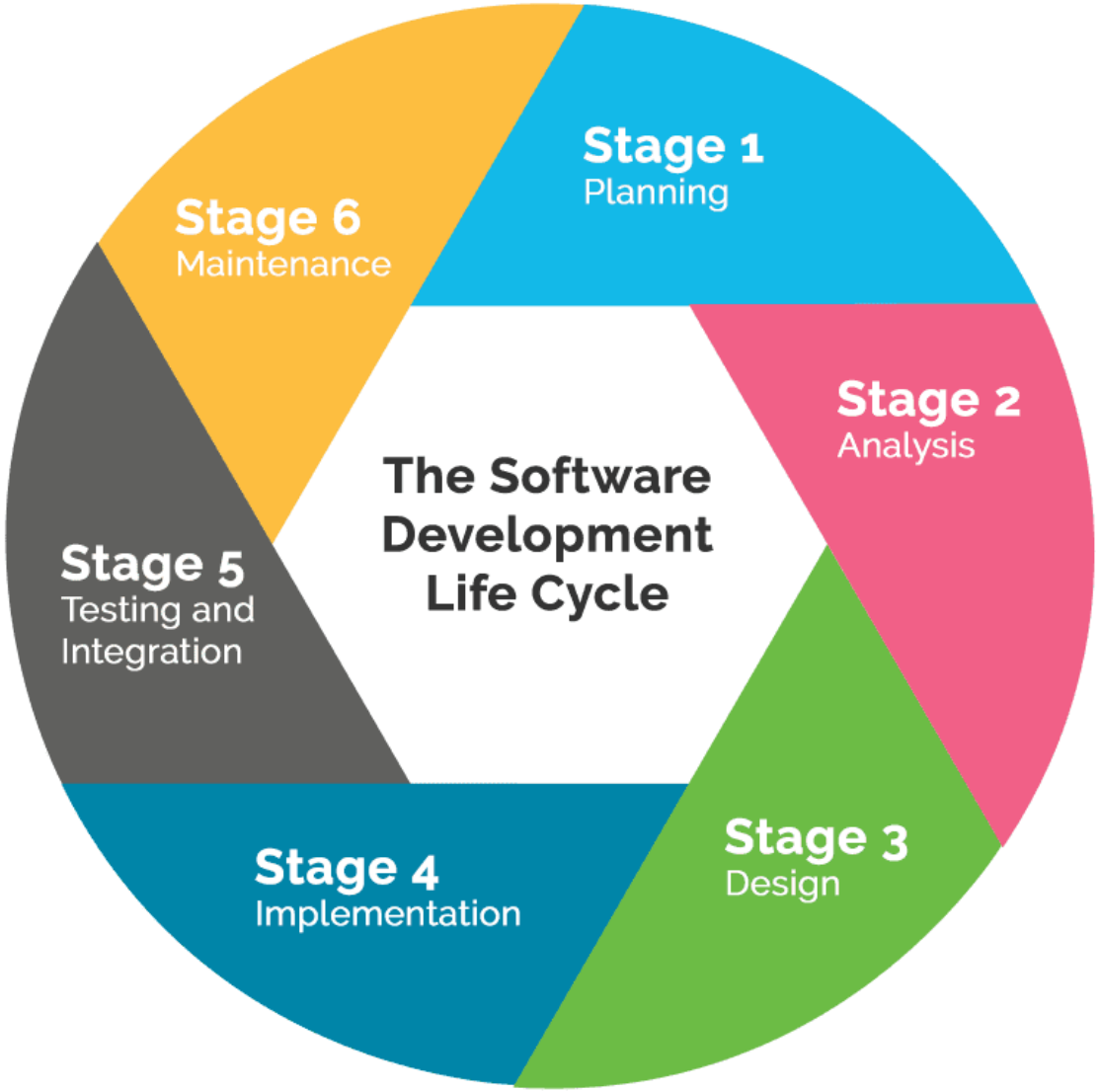
08 Boehm's Spiral Methodology

09 Incremental Software Development Methodology



Software Engineering Methodology

What is Software Engineering Methodology?

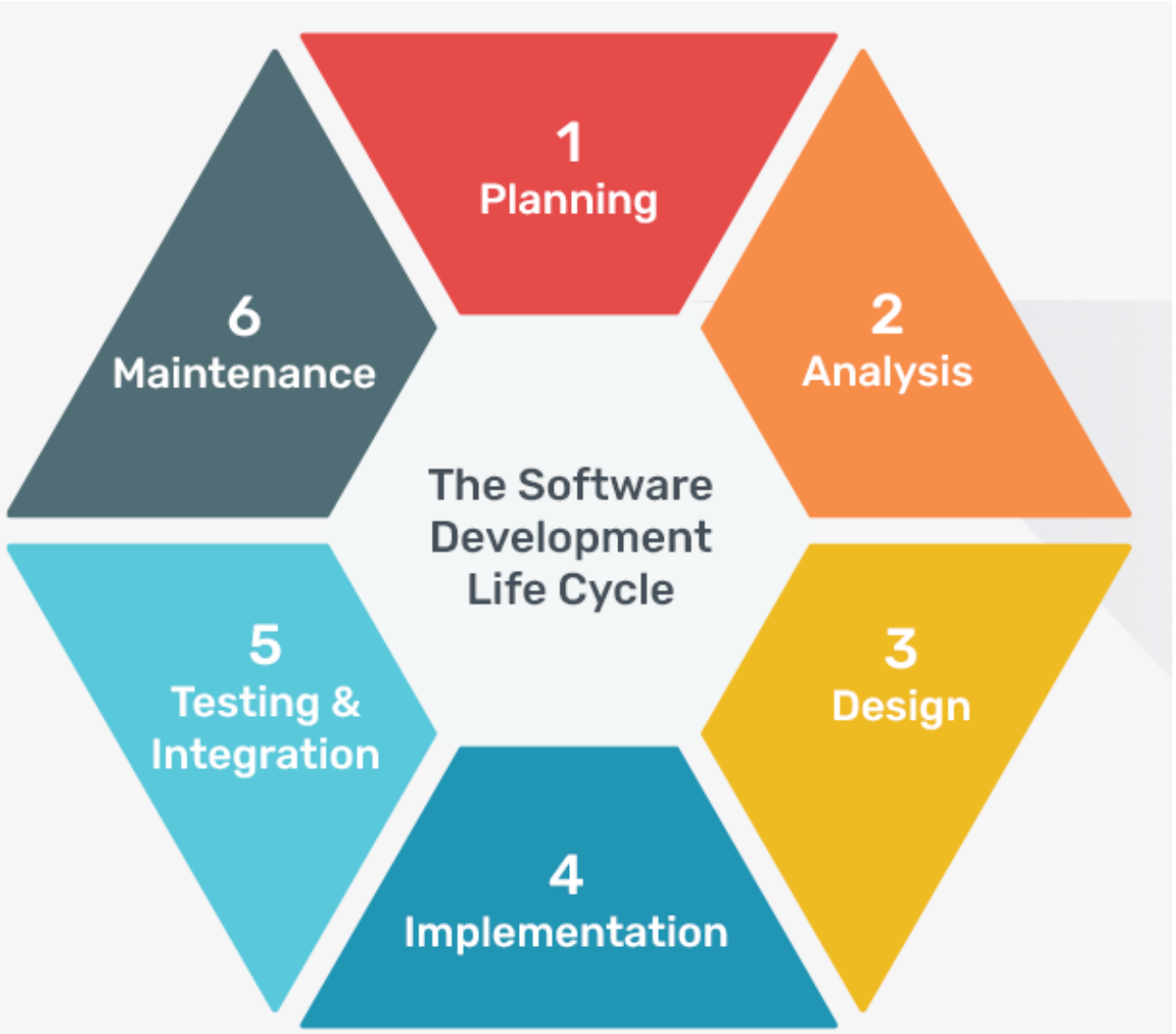


A **framework that is used to structure, plan, and control the process of developing an information system / software.**

What is Software Engineering Methodology?

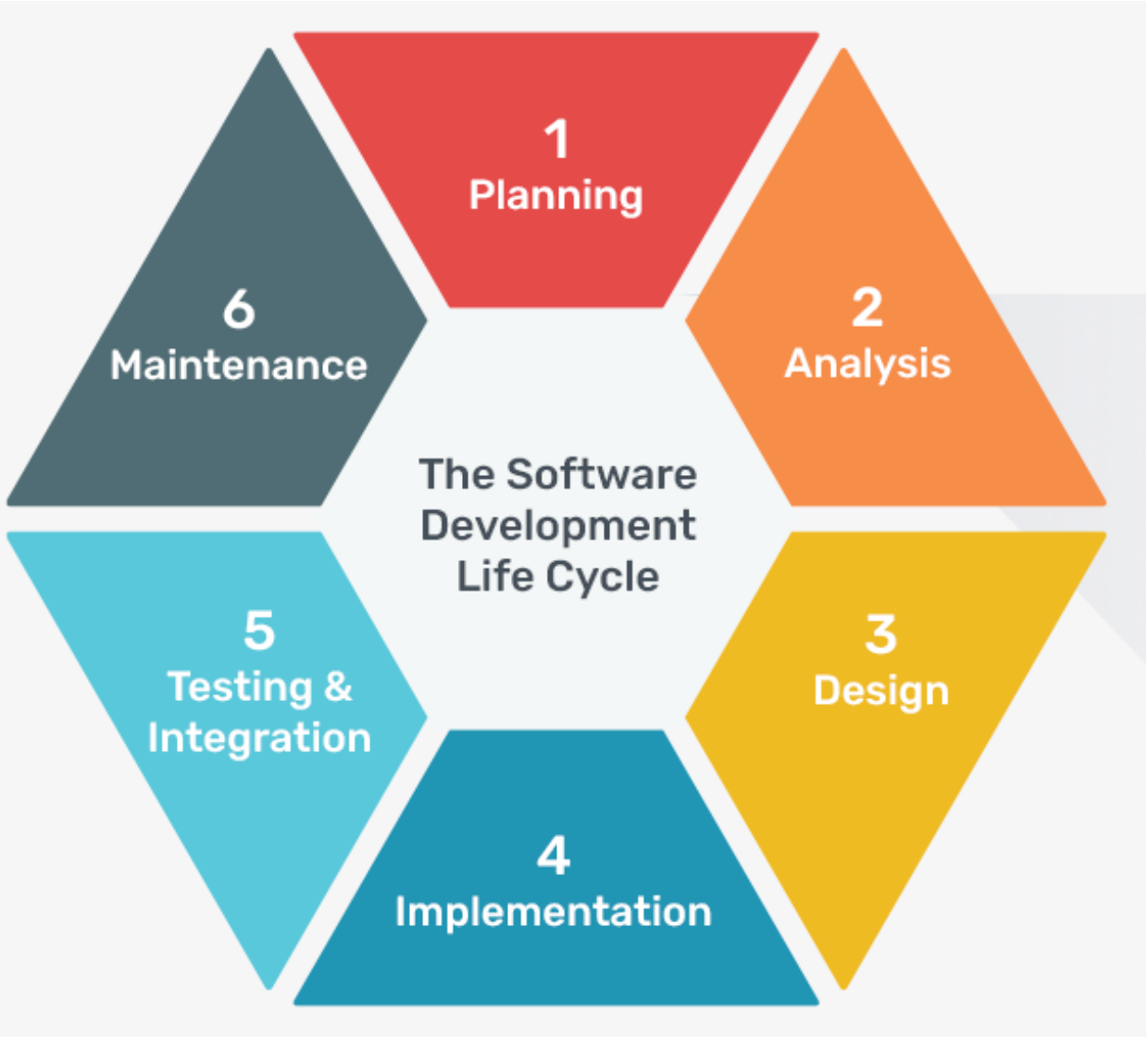
- ✓ A methodology is a way of making or adopting a model to perform a task or a set of tasks so that the goal of that task can be achieved as predicted.
- ✓ A methodology can also be defined as a single method or a set of methods through which a goal can be achieved.
- ✓ The decision of selecting a specific method or methods depends on the requirements of that method (or methods). A software product can be developed using either some well-known methodologies or any methodology that may be available to the developer.

Why a Methodology is Important?



In order to apply the engineering techniques to build a product, a methodology that has been proven successful and effective in terms of cost, time, quality, and so on is needed.

Why a Methodology is Important?



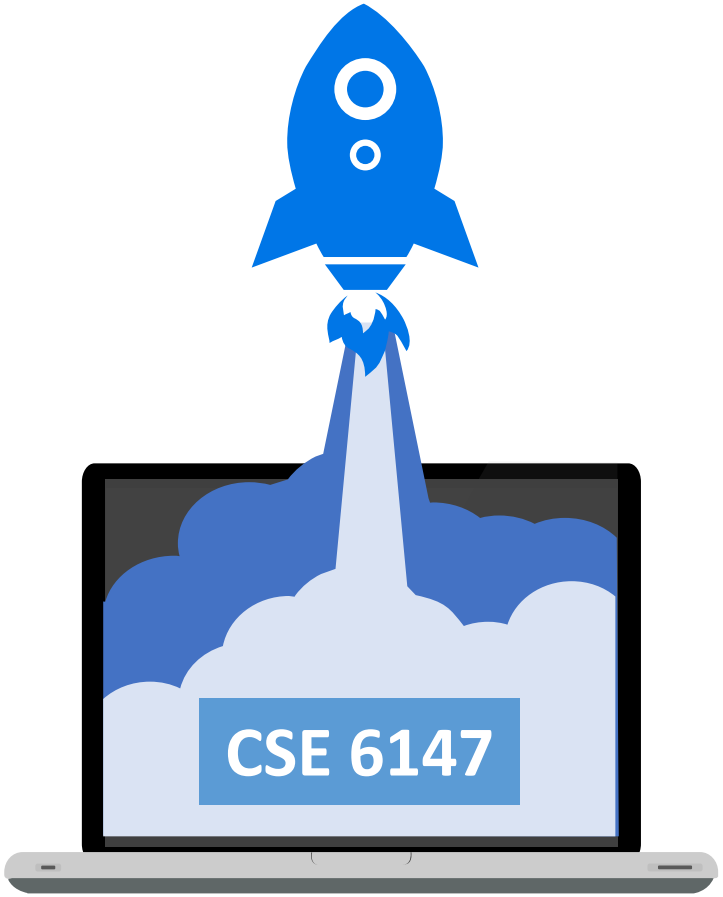
Adhering to a properly-defined methodology enables a project to provide better estimates, deliver stable systems, keep the customer informed, create a clear understanding of the task ahead, and identify pitfalls earlier, allowing for ample time to make adjustments.

Common Software Development Methodologies

- ☑ Agile Methodology
- ☑ Waterfall Methodology
- ☑ Rational Unified Process (RUP) Methodology
- ☑ Spiral Methodology
- ☑ Scrum Methodology
- ☑ eXtreme Programming Methodology

Common Software Development Methodologies

- ☑ Incremental Software Development Methodology
- ☑ Rapid Application Development Methodology
- ☑ Prototype Methodology
- ☑ Feature-driven Development Methodology
- ☑ Joint Application Development Methodology



Agile Methodology

What is Agile Methodology

- **Agile methodology is one of the most popular software development methodologies in recent days.**
- **It takes a different approach from the conventional, linear method.**
- **Agile focuses on how to satisfy the users/customers instead of emphasizing rigid procedures.**

What is Agile Methodology

With Agile, tasks are broken into short sprints that take about 1 to 4 weeks to complete. It's an iterative model

Pros and Cons of Agile

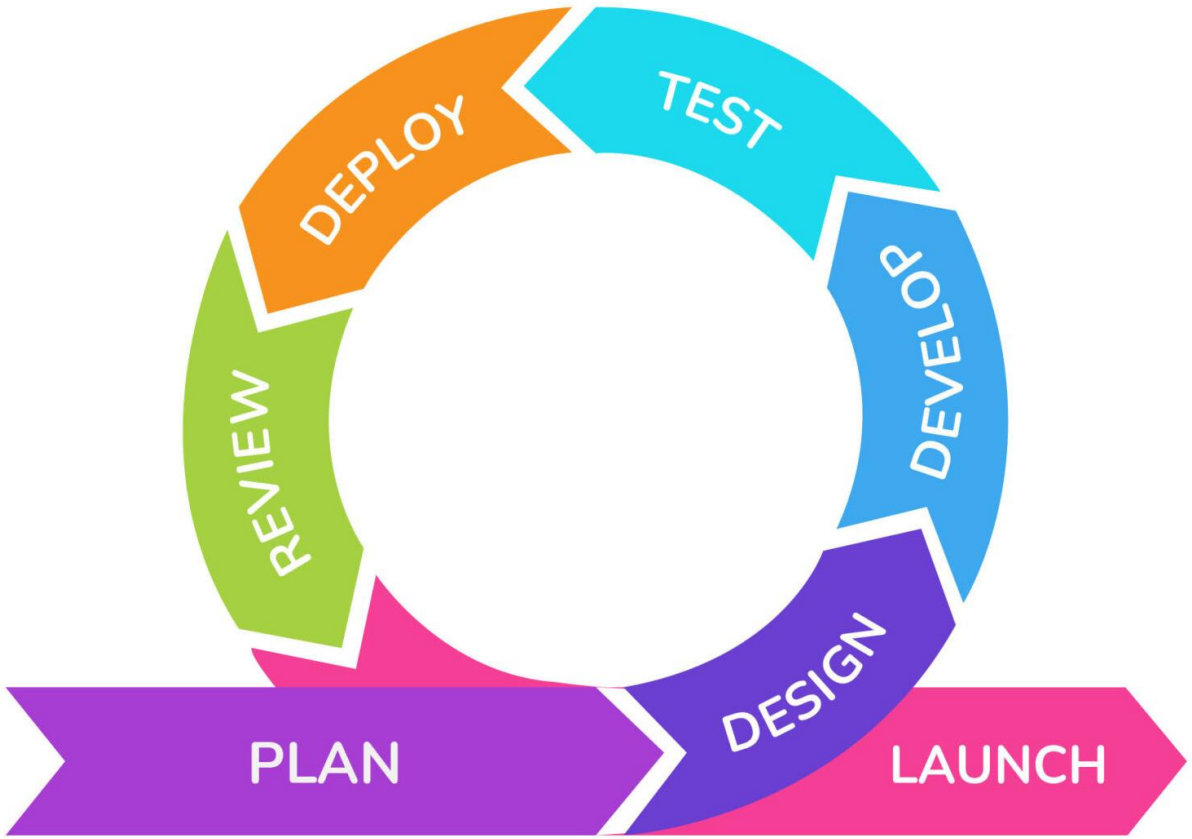
Advantages/Strengths/Pros

- ✓ Software has minimal defects due to the iterative effort in testing and fine-tuning.
- ✓ Clarity between team members during development, thanks to the frequent and transparent development.
- ✓ Changes in project requirements are easily addressed with little impact on the timeline.
- ✓ An overall improvement and **customer satisfaction** is achieved

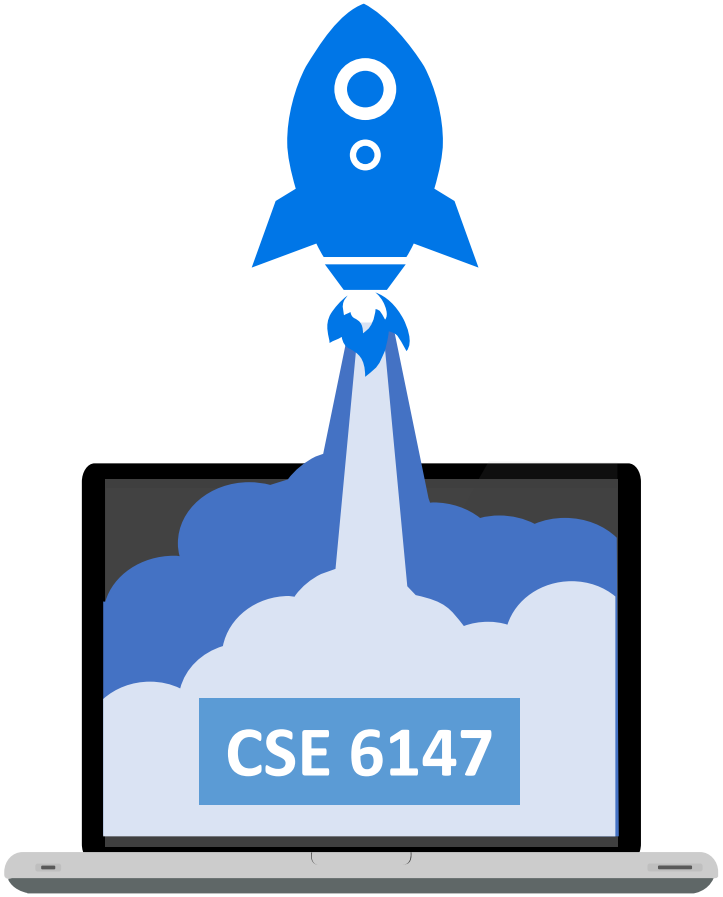
Disadvantages/Weaknesses/Cons

- 🚫 The team can sometimes lose focus due to overwhelming change requests.
- 🚫 Documentation takes a back seat in Agile, which can be a problem later
- 🚫 Agile focuses on discussions and feedback, which can be too time-consuming for the team.
- 🚫 Due to its non-structured approach, Agile requires **experienced developers** who can work independently.

When to Choose Agile Methodology?

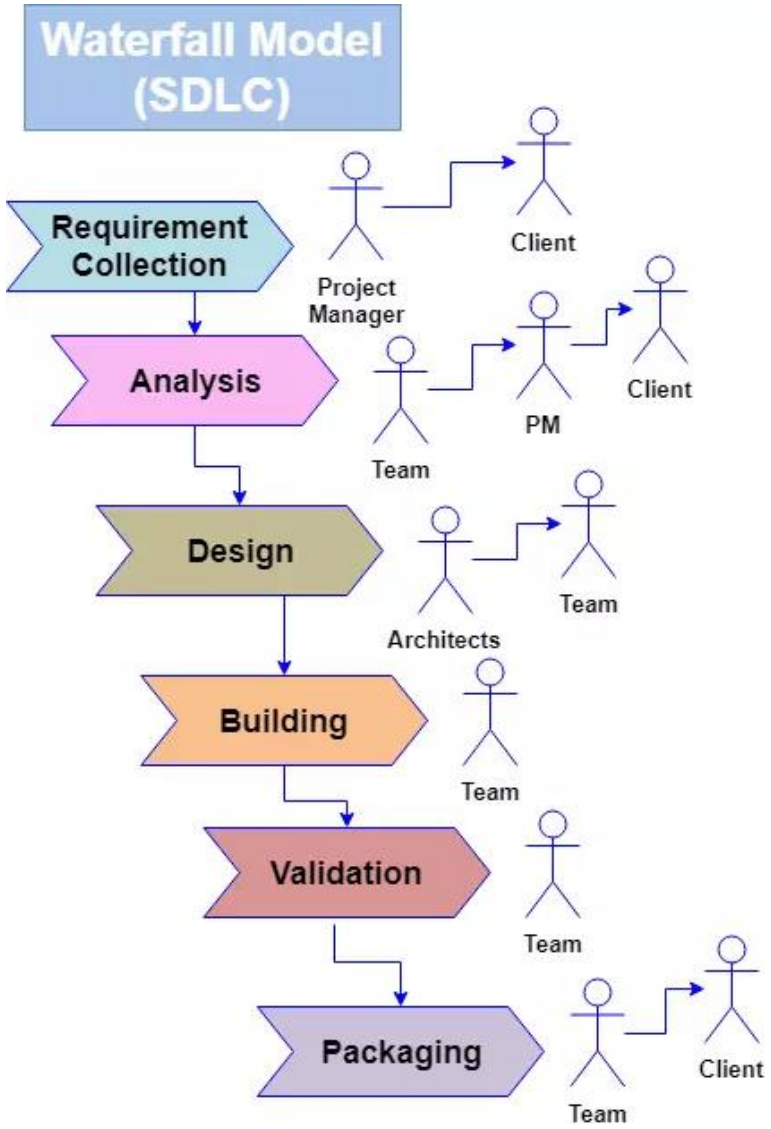


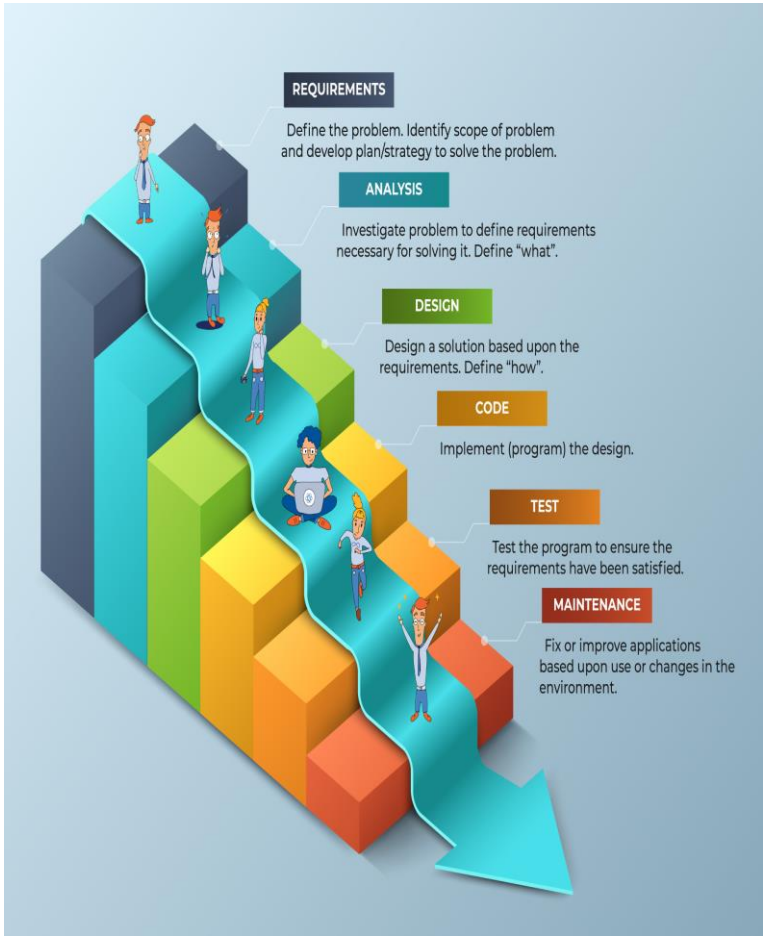
- Ideal for the projects with fast-changing or evolving requirements.
- A software in a new niche, want to use Agile.
- It can address the market needs.
- Assumption: The development team is skill full and able to work independently in a fast-paced and non-structured environment.



Waterfall Methodology

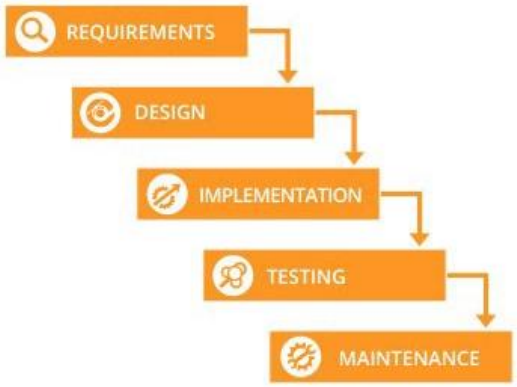
What is Waterfall Methodology

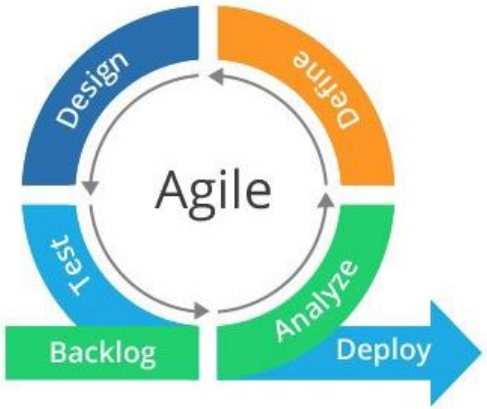




- In Waterfall methodology, developers complete one stage entirety before beginning the next.
- Each stage has its own requirements and plan and is reliant on inputs from the previous stage.
- There's no overlap of work between any two stages.

Agile vs Waterfall Methodology





PROJECT TRAIT/FACTOR	Agile	Waterfall	Comment
Customer Availability	Availability Throughout the project	At end of the project	Involvement reduces risk
Scope/ Features	Ready for any instant changes in project	At end or as per the contract singed by the customer	Terms and conditions make restrictions
Feature Prioritization	Work on valuable features first to implement	Worked as pe the document singed and give everything which mentioned in document	Do everything to get successful result

Pros and Cons of Waterfall

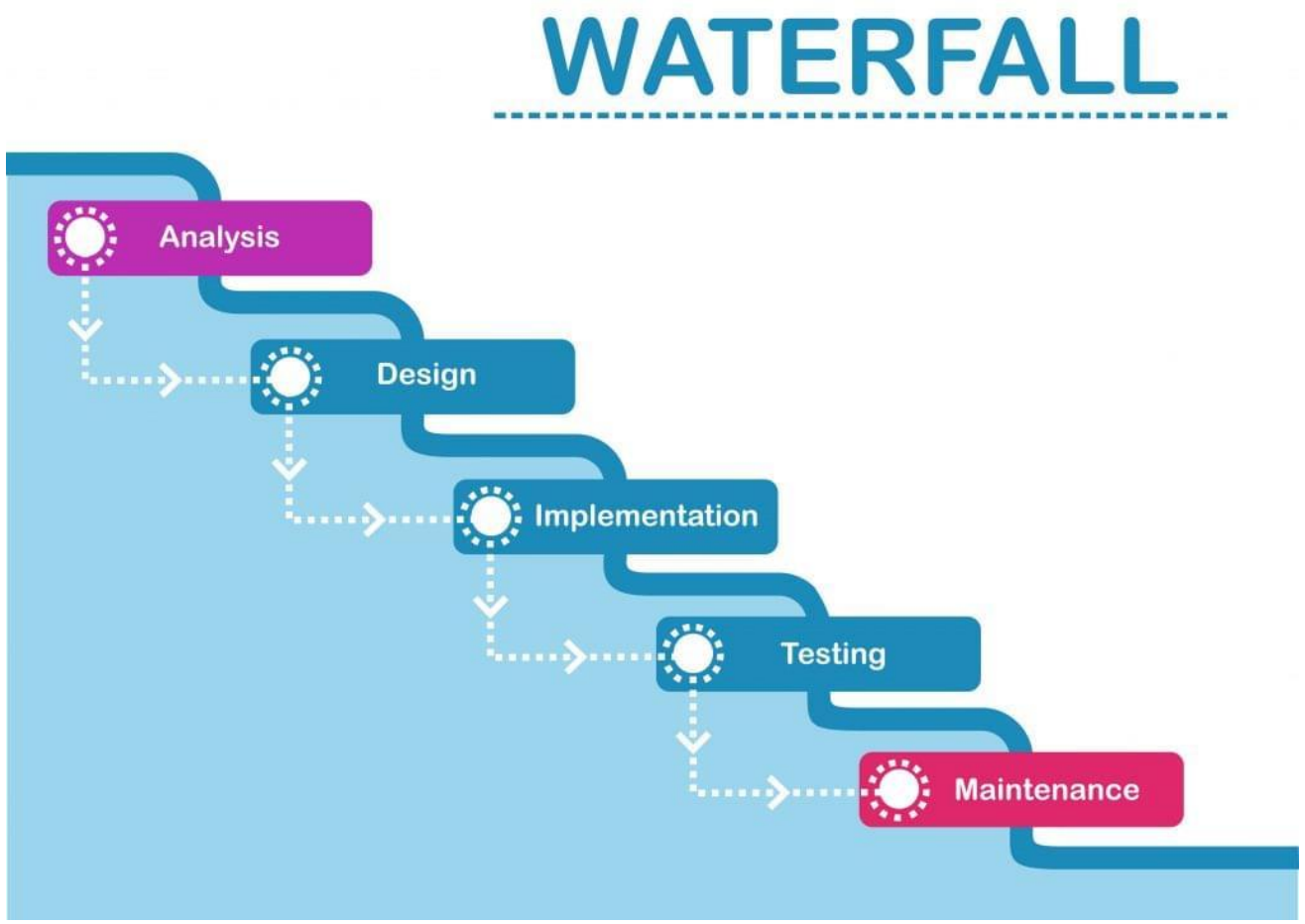
Advantages/Strengths/Pros

- ✓ The linearity of the waterfall model makes it easy to understand, particularly for **new developers.**
- ✓ All specifications and deliverables are spelled out **before the development commence.**
- ✓ Less room for miscommunicating as it's clearly defined in each stage.

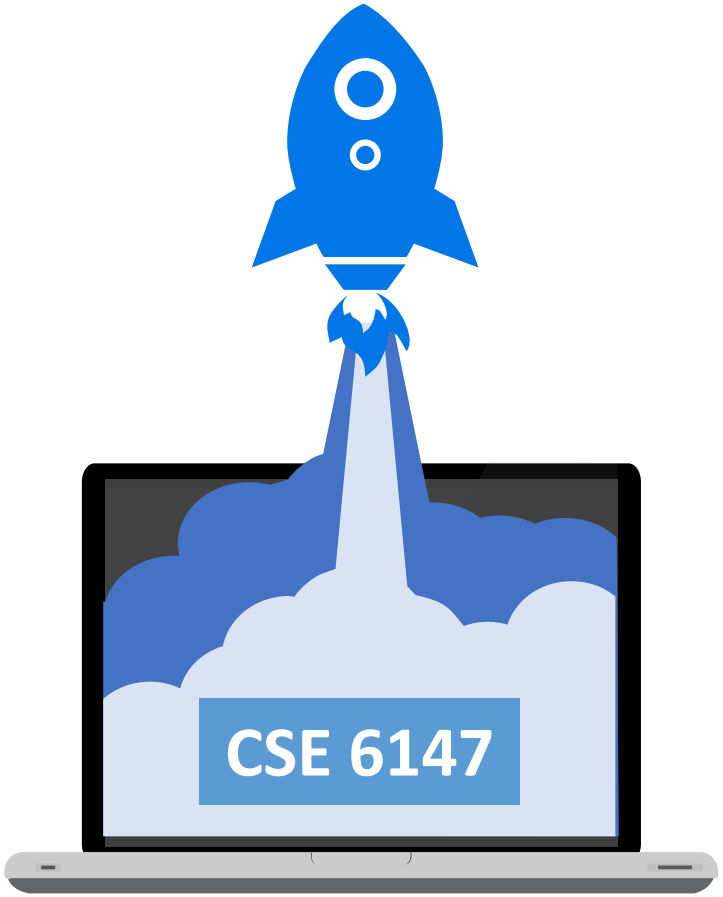
Disadvantages/Weaknesses/Cons

- 🚫 It doesn't include customer feedback in the early phases, which increases the risk of the project.
- 🚫 Testing is only executed at the end of the development. Some problems are harder to fix at a later stage.
- 🚫 The rigidity of the waterfall model gives no room for changes, **making it unsuitable for complex projects.**
- 🚫 The team can spend **too much time on documentation** instead of delivering solutions that solve the user's problems.

When to Choose Waterfall Methodology?

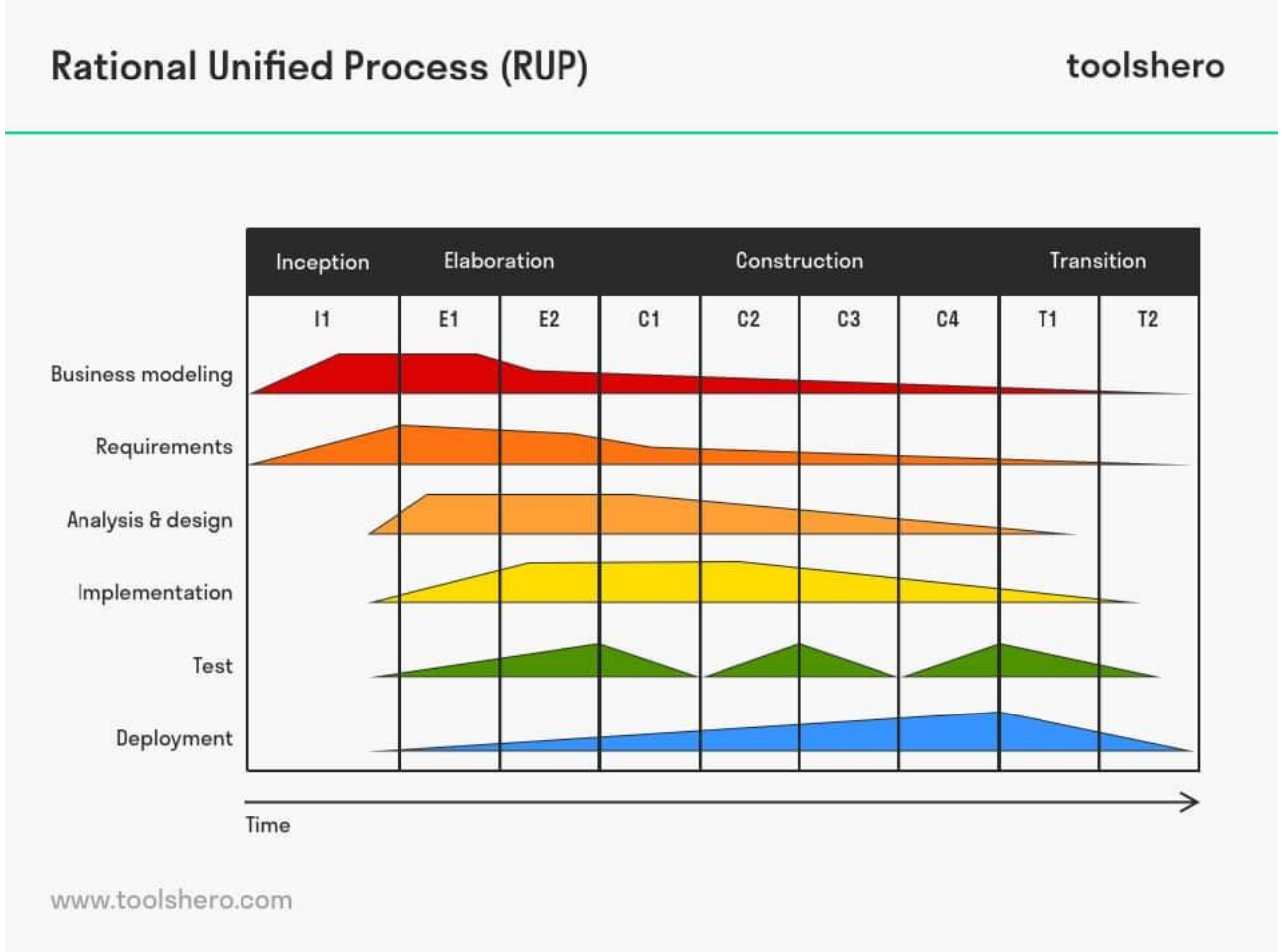


- Use waterfall only when you have a project with clearly-defined scope.
- It is **not suitable** for development that involves many unknowns.
- Waterfall is ideal for projects with predictable outcomes and when you have a team of inexperienced developers.



Rational Unified Process (RUP) Methodology

What is Rational Unified Process (RUP) Methodology



- Rational Unified Process (RUP) is an agile software development method, in which the life cycle of a project, or the development of software, is **divided into four phases:** **M**odelling, **A**nalysis and Design, **I**mplementation, **T**esting

Pros and Cons of RUP

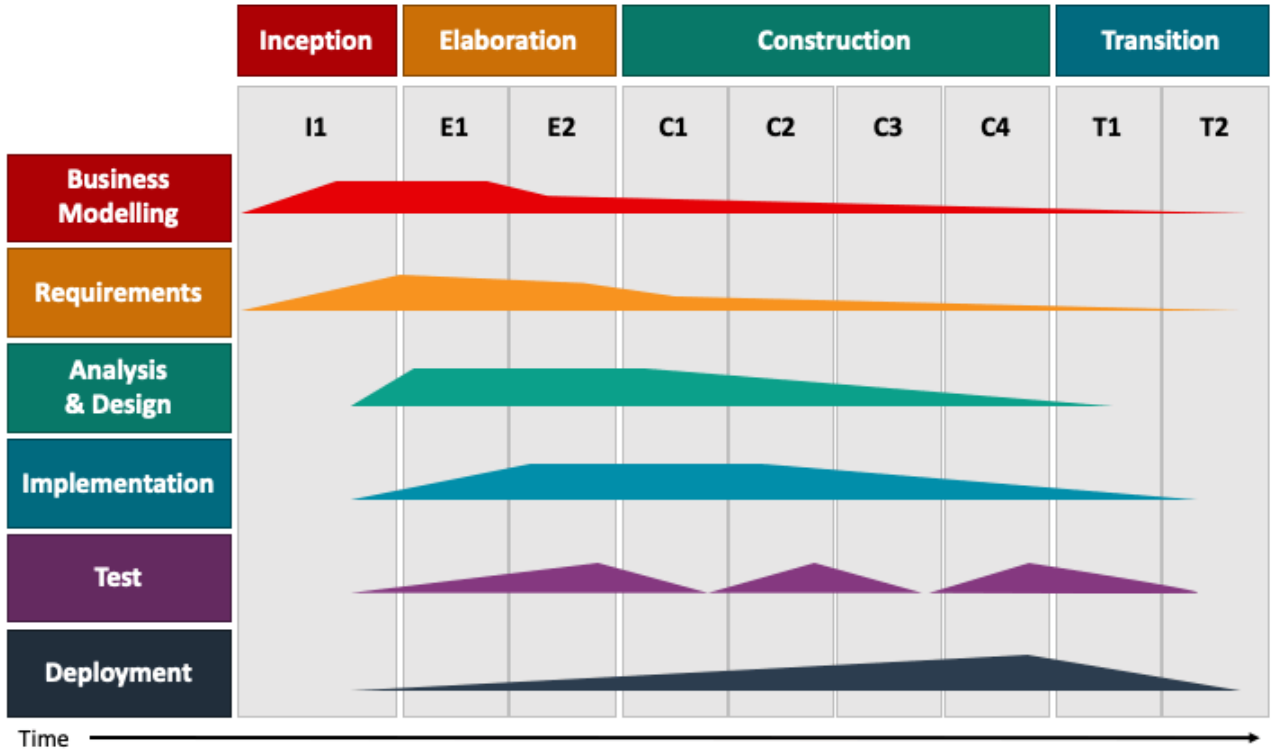
Advantages/Strengths/Pros

- ✓ It provides good documentation, it completes the process in itself.
- ✓ It provides risk-management support.
- ✓ It reuses the components, and hence total time duration is less.
- ✓ Good online support is available in the form of tutorials and training.

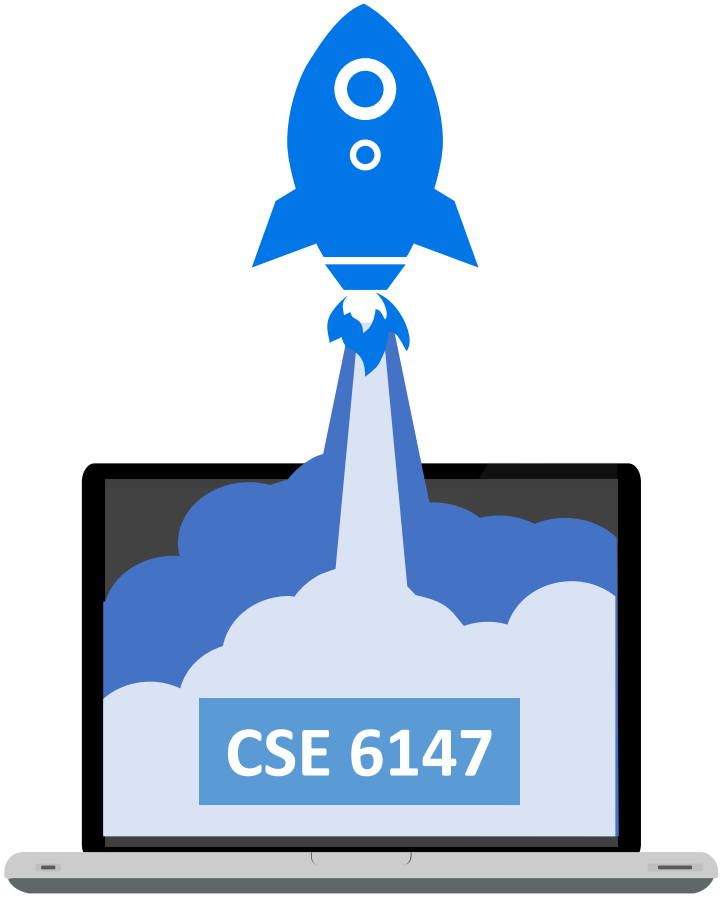
Disadvantages/Weaknesses/Cons

- 🚫 Team of expert professional is required, as the process is complex.
- 🚫 Complex and not properly organized process.
- 🚫 More dependency on risk management.
- 🚫 Hard to integrate again and again.

When to Choose RUP Methodology?

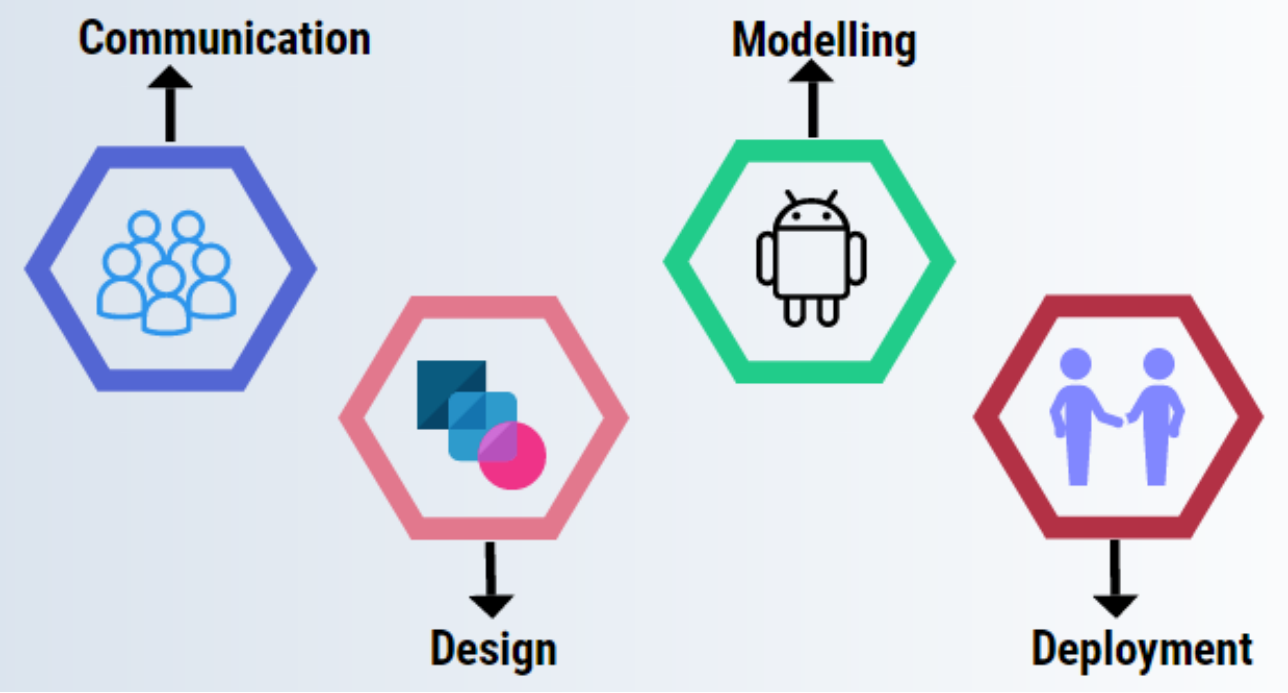


- Due to it's incremental flexibility RUP is used more often for the development of large, complex, engineered systems.



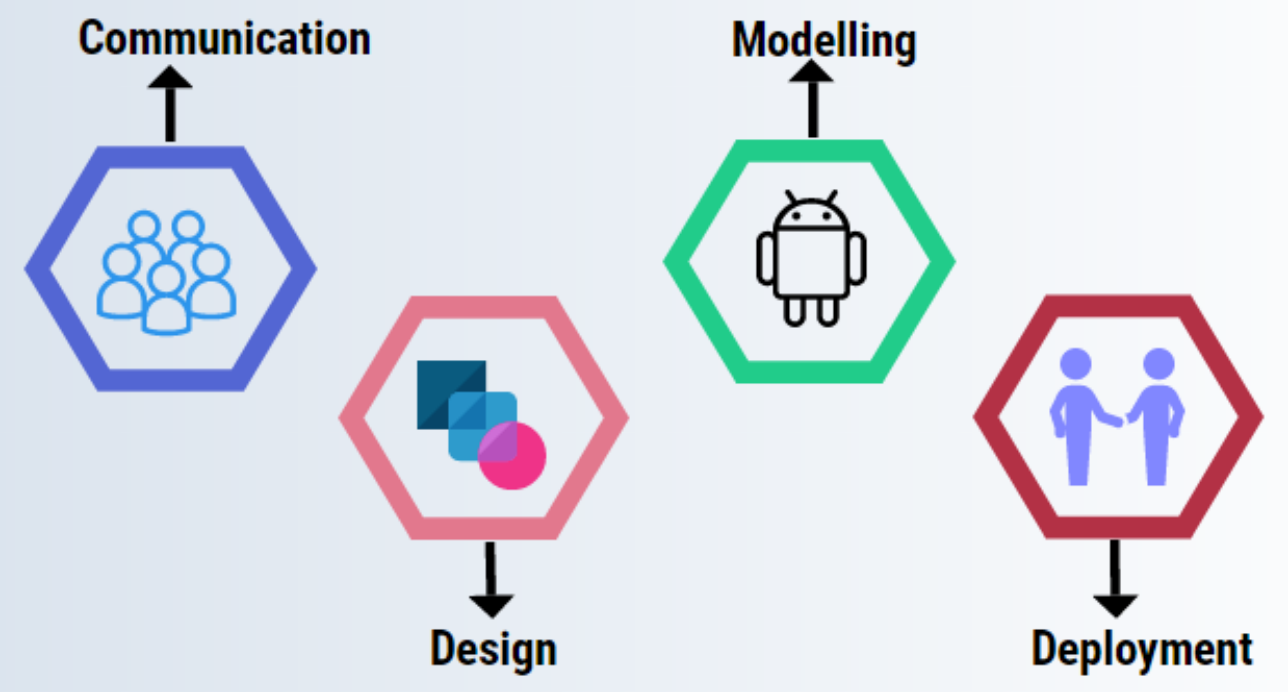
Prototype Methodology

What is Prototype Methodology



- Instead of developing a full-fledged software, the prototype model allows developers to work on the prototype version of the final product.
- The prototype is then made available for customer testing, evaluation, and feedback.

What is Prototype Methodology



- Based on the gathered feedback, the prototype goes through several iterations of refinement **until it's deemed satisfactory** by the customer.
- The appeal of the prototype approach is its **rigorous evaluation that uncovers possible issues** before actual development begins.

Pros and Cons of Prototype

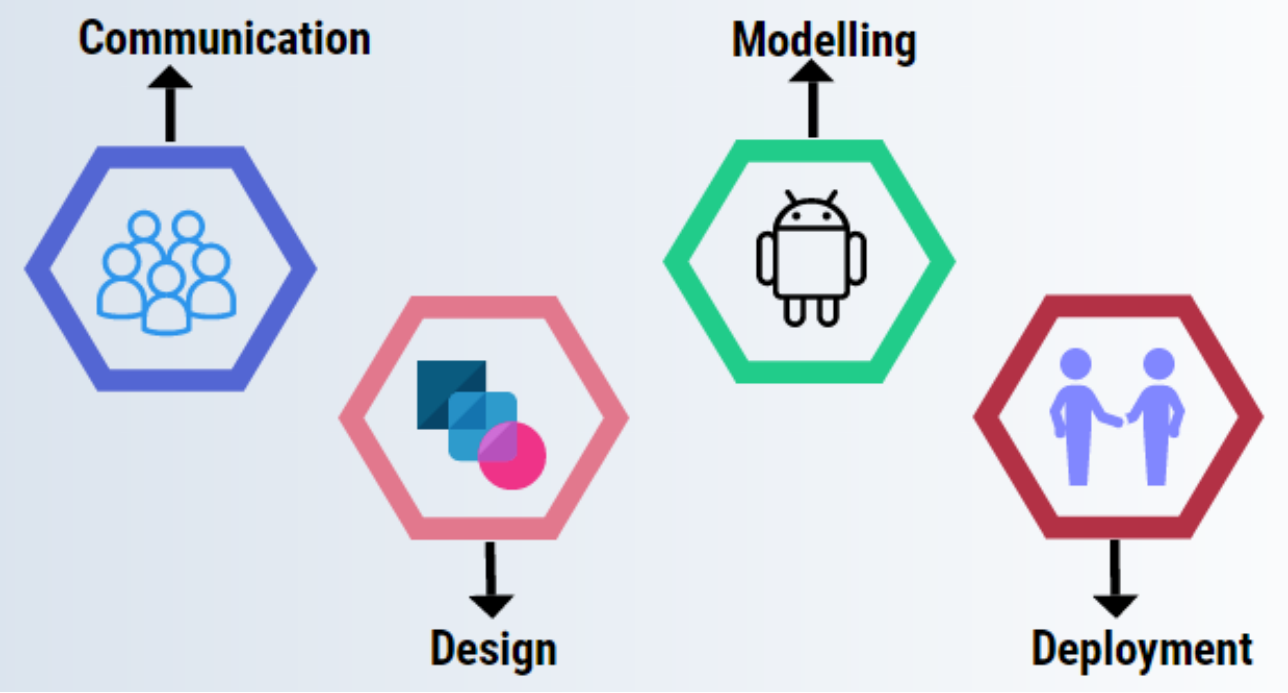
Advantages/Strengths/Pros

- ✓ Good in ironing out potential issues in the early development stage, which greatly reduces product failure risk.
- ✓ Ensures the customer is happy with the 'product', before real development started.
- ✓ Build rapport with the customer early on with the discussions, which helps throughout the project.
- ✓ Gather detailed information with the prototype, which is later on used in building the final version.

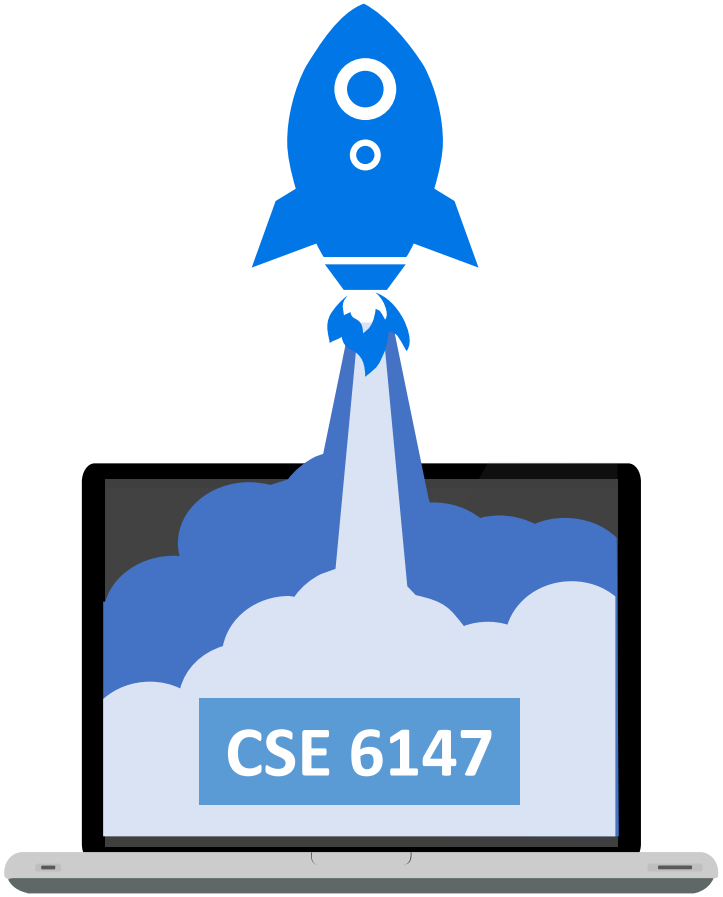
Disadvantages/Weaknesses/Cons

- 🚫 Excessive to and from in testing out the prototype with the customer can delay the development timeline.
- 🚫 The customer's expectations of the actual product may not align with the prototype.
- 🚫 There's a risk of cost overrun as the works on the prototype are often paid for by the developer.

When to Choose Prototype Methodology?

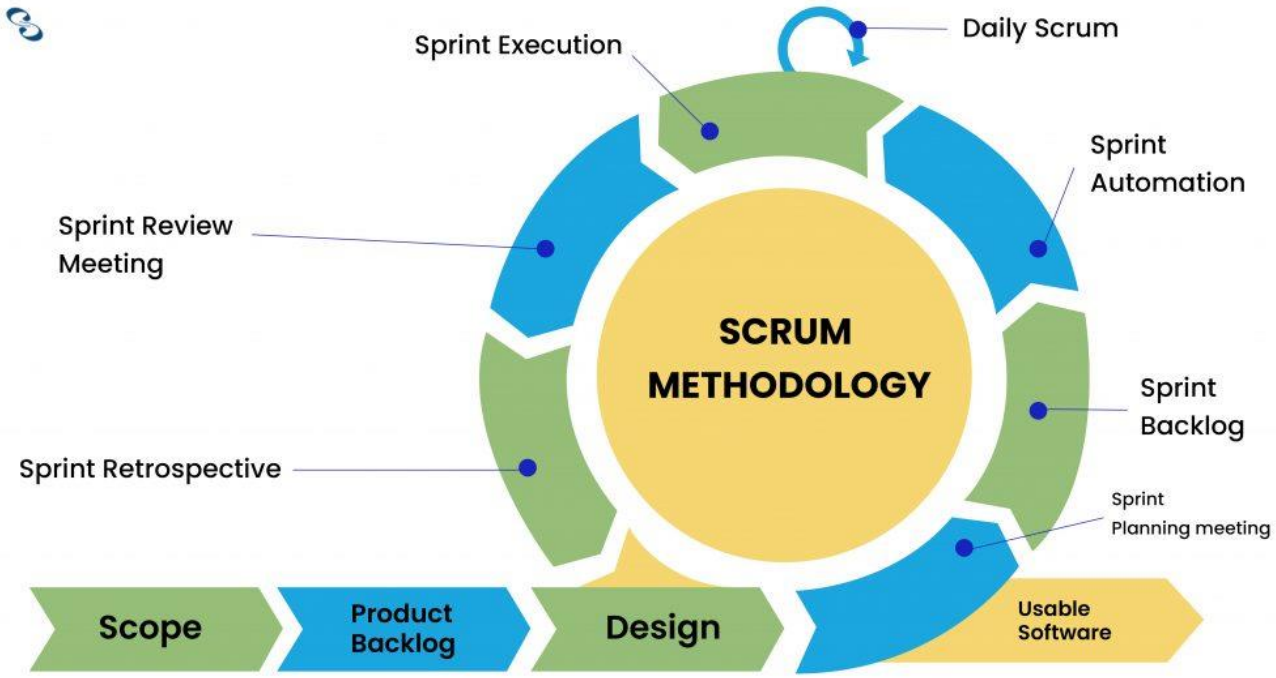


- The prototype model is ideal when you're building software with many unknowns. For example, an online platform with intense user interaction.
- To find out what works best with users and reduce the risk of developing the actual product.



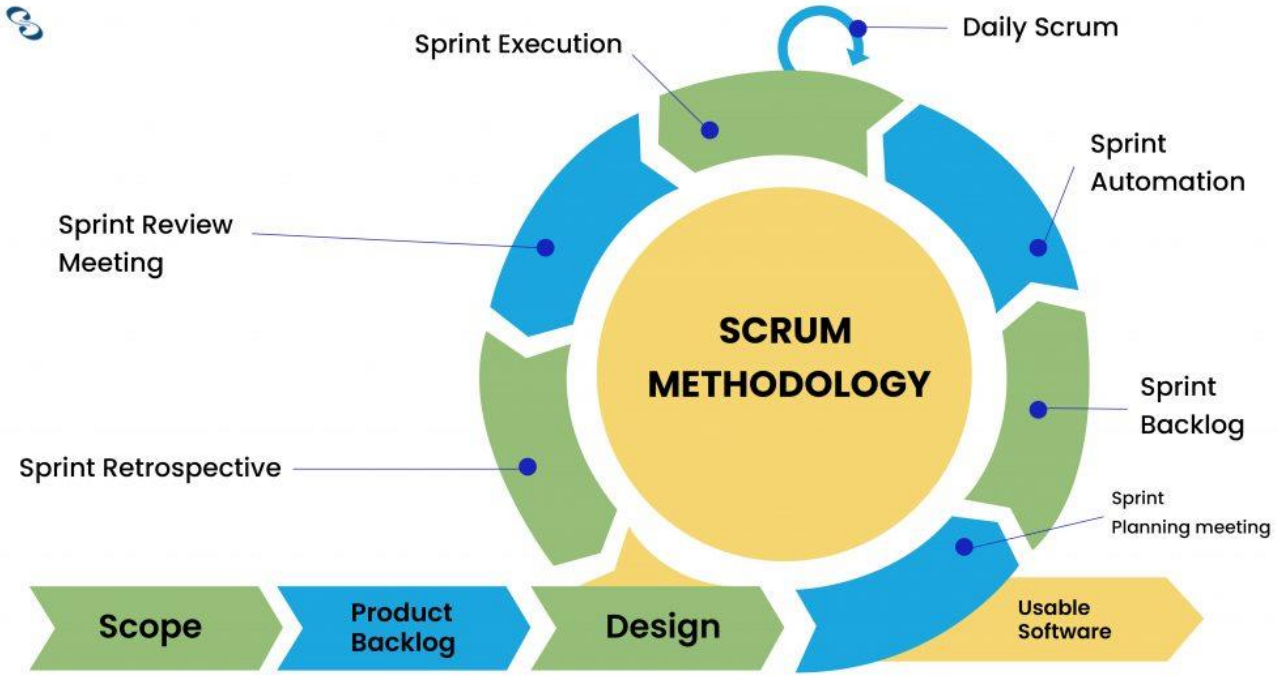
Scrum Methodology

What is Scrum Methodology



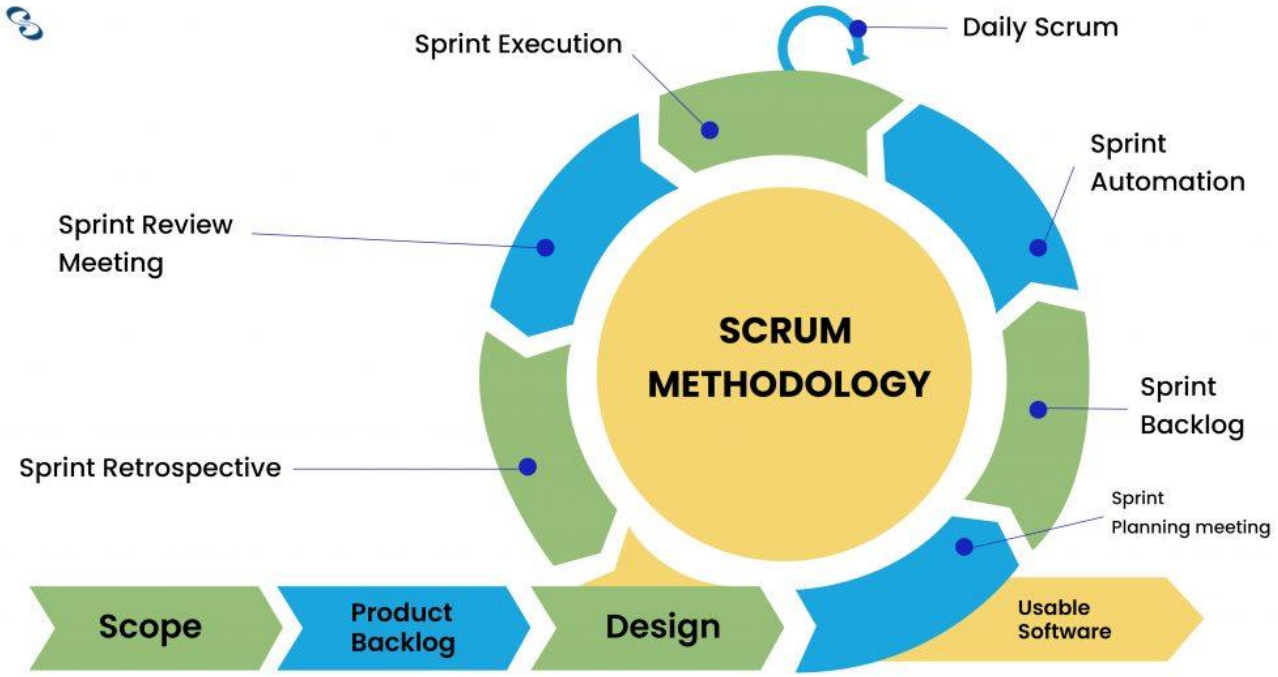
Scrum is arguably one of the most flexible software development methodologies available. It is based on the **Agile** philosophy and is favoured for **its incremental and iterative approaches**. The Scrum methodology involves the **Product Owner, Scrum Master, and the Development Team.**

What is Scrum Methodology



The **product owner** takes input from the client and ensures that the team is on track in fulfilling the client's requirements. Meanwhile, the **Scrum Master** acts as a facilitator and ensures that team members are familiar with the Scrum process. The **team** takes charge of executing the development.

What is Scrum Methodology



What makes Scrum an ideal methodology in a fast-paced environment is how tasks are executed in sprints. **Each sprint takes up to 4 weeks.** The speedy execution allows teams to identify issues, introduce solutions, test, and gather feedback in a short period. It makes tackling fast-paced projects much easier.

Pros and Cons of Scrum

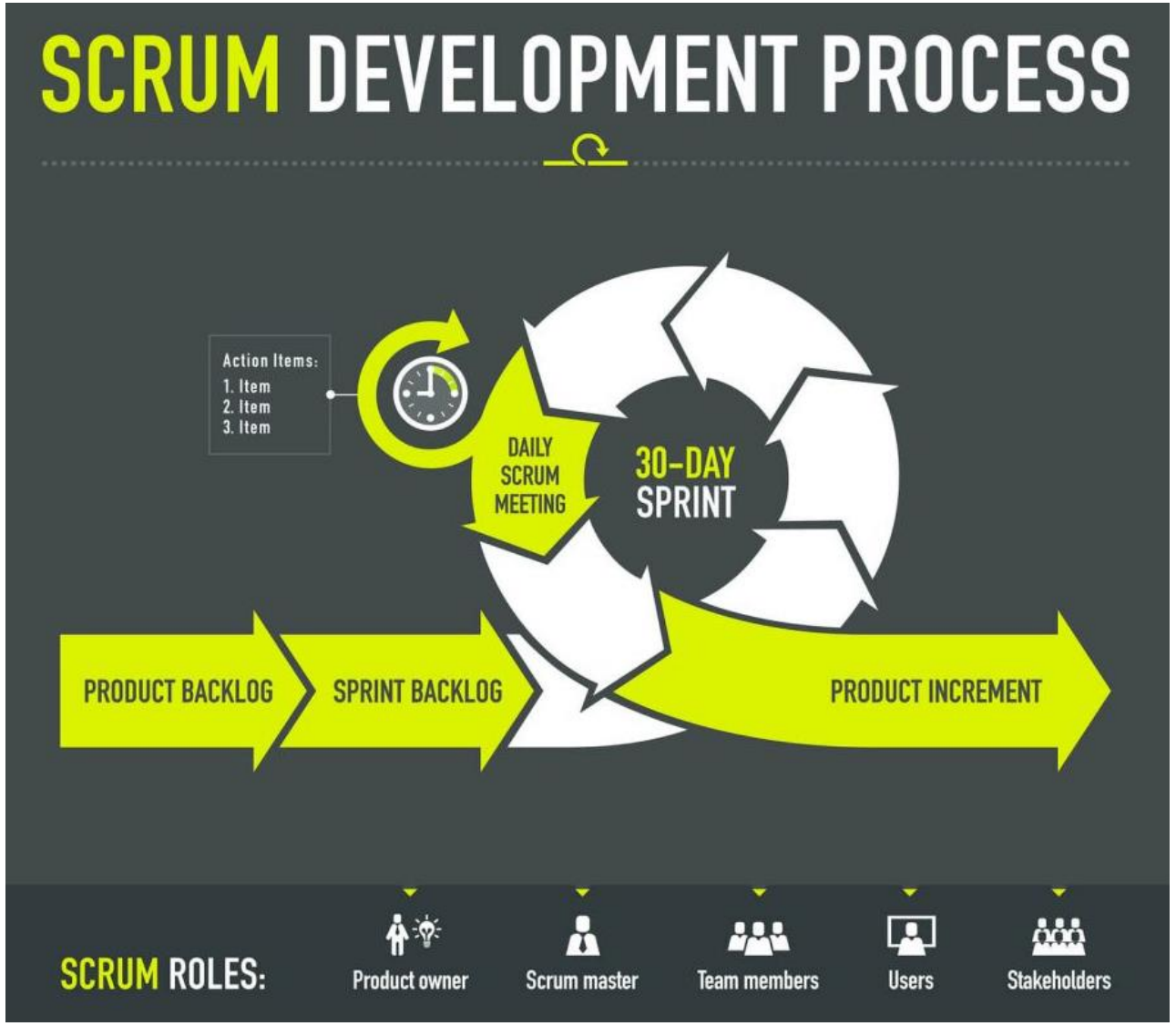
Advantages/Strengths/Pros

- ✓ Short iterations allow quick resolutions to problems.
- ✓ Scrum is very responsive to changes as the process includes regular feedback.
- ✓ Scrum is economical and effective.
- ✓ **Regular meetings ensure that team members are on the same page at all times.**
- ✓ Contributions of individual members are noticed and appreciated through the Scrum meetings.

Disadvantages/Weaknesses/Cons

- 🚫 **All team members must be equally skilled and committed for Scrum to work.**
- 🚫 The daily Scrum meetings can be draining for team members.
- 🚫 May increase time-to-market if there's no strict control on the deadline.
- 🚫 **Not suitable for large projects.**

When to Choose Scrum Methodology?



Scum is the go-to methodology if you have a project with vague requirements but needs to adapt to frequent changes. For example, you need to get an MVP built quickly and test it out amongst users. Remember that **Scrum is only effective if you have a committed and experienced team.**

Thank You