



SEN319 Software Project Management (Fall 2023)

Project Management Methodologies and Life Cycle of Projects

Asst.Prof.Dr. Hasan ÇİFCİ



Agenda

- **Project Management Methodologies**
- **Software Development Methodology Frameworks**
- **SDLC Methodologies in Detail**
- **Factors to Choose a Methodology**
- **Waterfall**
- **Prototyping**
- **Spiral Model**
- **V-Shaped Model**
- **Incremental Development**
- **The Agile Family**
- **PMBOK**
- **PRINCE2**



Introduction

Project management methodology

Systems development life cycle

Software development life cycle (SDLC)

System:

- Definition is dependent of the domain you aimed.
- Cultural, sociological or engineering domains have their definition.
- In computer science and information science, **system** is a hardware system, software system, or combination, which has components as its structure and observable inter-process communications as its behavior.

Project Management Methodologies

- A project management methodology is a system of guided procedures for managing a project. In other words, it's a set of rules that you play by to get your project finished in the most productive way possible.



- The reason there are so many different project management methodologies is because there are different kinds of projects and different kinds of teams.





Project Management Methodologies Groups (Approach-1)

- There are basically two major kinds of projects. One is generally called “**Predictive**” and the other is called “**Adaptive**”.
- A **predictive** lifecycle is where the requirements are clear in the beginning of the project, hence documented, converted to specifications and then executed and after execution the scope is validated to ensure that all the requirements have been met.
 - This kind of lifecycle assumes that there would not be many changes to the scope of the project during the course of the project.
- The **adaptive** lifecycles (like agile) are undertaken because the scope of the project is not very clear or fixed at the early stage of the project.
 - The journey itself is expected to unravel more requirements. Which means that these lifecycles are open to “changes”.



Project Management Methodologies Groups (Approach-2)

1. Sequential Methodologies:

Waterfall

Critical Path Method (CPM)

Critical Chain Project Management (CCPM)

2. The PMI/PMBOK “Method”

3. Agile Methodologies

Scrum Project Management

Kanban

Extreme Programming (EP)

Adaptive Project Framework (APF)

4. Uncertainty Methodologies

Event Chain Methodology (ECM)

5. Process-Based Methodologies

PRINCE2

Lean Project Management

Six Sigma



Project Management Methodologies Groups (Approach-3)

At a bare minimum, we can characterize processes along two main dimensions:

1. The linearity of the process: The order in which the development of more elementary components of the project is organized.

- a. Sequential development**
- b. Cyclical development**
- c. Parallel development**

2. The formality of the process: The amount of infrastructure a project requires.

- a. Traditional Project Management (PM)**
- b. Agile PM**
- c. Extreme PM**



Project Management Methodologies Groups (Approach-3)

The Linearity of the Process

- **Sequential development:** Proceeds from specification to implementation, with little opportunities for backtracking.
- **Cyclical development:** Organizes development in different rounds, with each round delivering more or improved functionality.
- **Parallel development:** Uses concurrent development. An initial activity organizes further development efforts, which are then carried out in independent and parallel tracks. A final activity integrates the contribution of the different tracks.



Project Management Methodologies Groups (Approach-3)

The Formality of the Process

- **Traditional PM:** The underlying assumption is that efficiency can be achieved with a top-down and planned organization of work.
- **Agile PM:** Focuses on efficiency and flexibility. For the supporters of agile methodologies, management is an infrastructure that adds unnecessary work and unnecessary rigidity to the process. According to their supporters, projects should exploit any opportunity to improve the quality of a product and the efficiency of its development.
- **Extreme PM:** High-risk and exploratory projects require even more flexibility than agile methodologies can provide. Extreme project management thus denotes a situation in which long-term planning is impossible: high speed, high change, and high uncertainty are the three conditions characterizing these projects. *(Note: In some sources, Extreme PM = Extreme Programming)*



Software Development Methodology Frameworks

- **Software Development Methodology Frameworks**
 - = Process models
 - = Software development life cycle (SDLC) methodologies
 - = Software development processes
- **SDLC** is a process by which software is developed and deployed.
- SDLC is a division of software development work into distinct phases containing activities with the intent of better planning and management.
- The methodology may include the pre-definition of specific deliverables and artifacts that are created and completed by a project team to develop or maintain an application.



SDLC Methodologies in Detail

- **Waterfall (Linear Sequential Model)**
- **Prototyping**
- **Spiral Development**
- **V-Model**
- **Incremental Development**
- **Agile Process Models** →
 - Scrum
 - Kanban
 - Lean Development
 - Extreme Programming (XP)
 - Adaptive Software Development (ASD)
 - Feature Driven Development (FDD)
 - Agile Modeling (AM)
 - Dynamic Systems Development Method (DSDM)
 - Crystal



Factors to Choose a Methodology

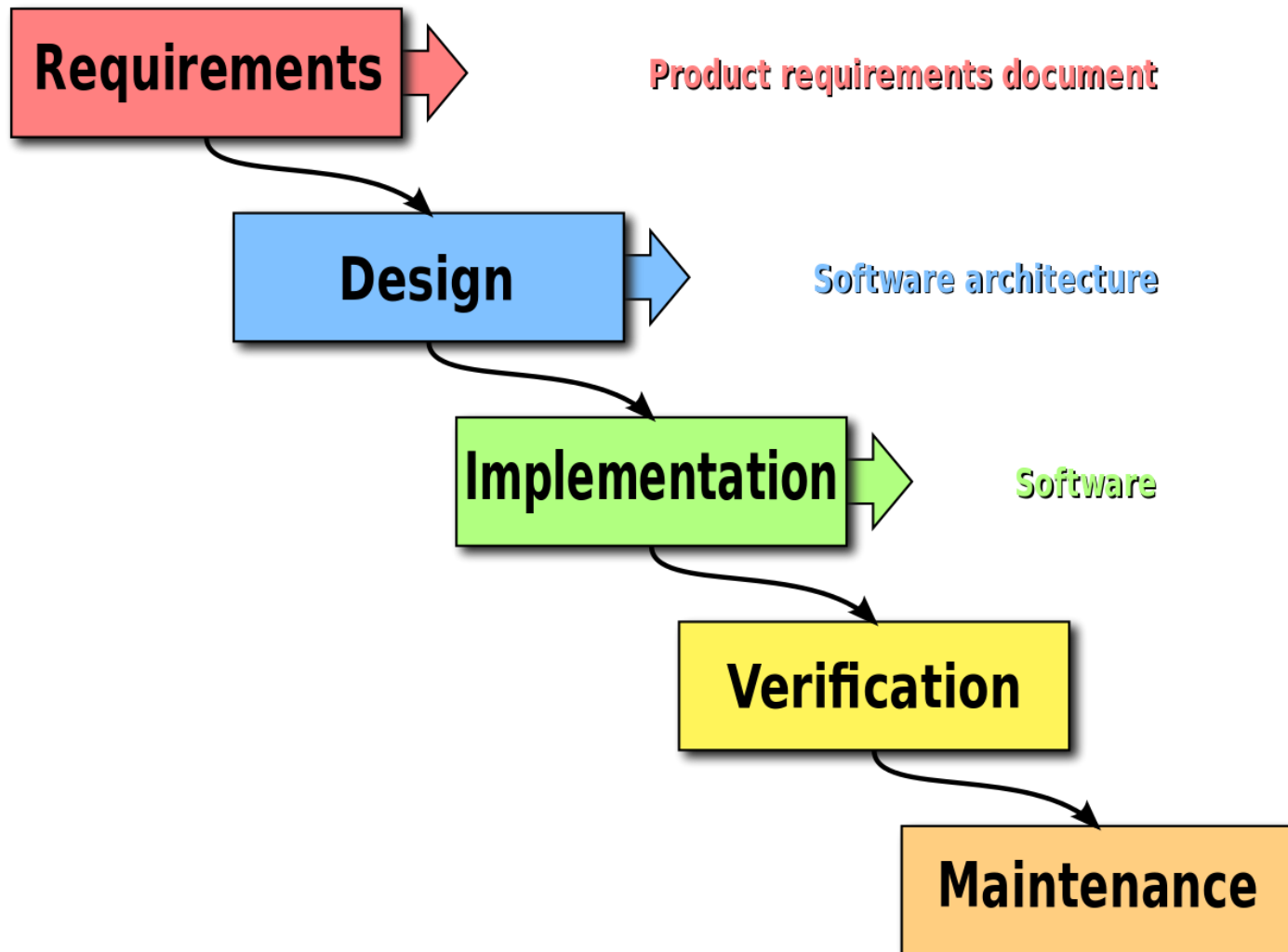
- Requirements certainty
- **Team's knowledge, skill and size**
- **Organizations and people involved**
- **Stakeholders' needs**
- **Size and complexity of software**
- **Criticality of the application**
- **Uncertainty and unpredictability of the project environment**
- **Regulations**
- Compliance with the selected technology
- Type of project
- Project risk and quality insurance



Waterfall

- Initially articulated by Dr. Winston Royce in 1970 to address the growing complexity of software development.
- Got wide acceptance in many industries but especially in the software industry.
- It is sequential.
- It is also heavily requirements-focused.
- You need to have a crystal-clear idea of what the project demands before proceeding further.

Waterfall





Waterfall

Advantages

- **Easy of use.**
- **Rigidity of the structure.**
- **Documentation.**
- **Efficiency.**
- **Well defined stages.**
- **Helps to plan and schedule the project.**
- **Specific deliverables at each phase.**
- **Early detection of errors/misunderstanding because of verification at each stage.**



Waterfall

Disadvantages

- It has higher risk since the methodology is rigid when you find error or need to change something.
- Assumes that the requirements of a system can be frozen.
- Very difficult to go back to any stage after it finished.
- A little flexibility and adjusting scope is difficult and expensive.
- Costly and required more time, in addition to the detailed plan.



Waterfall

When to Choose?

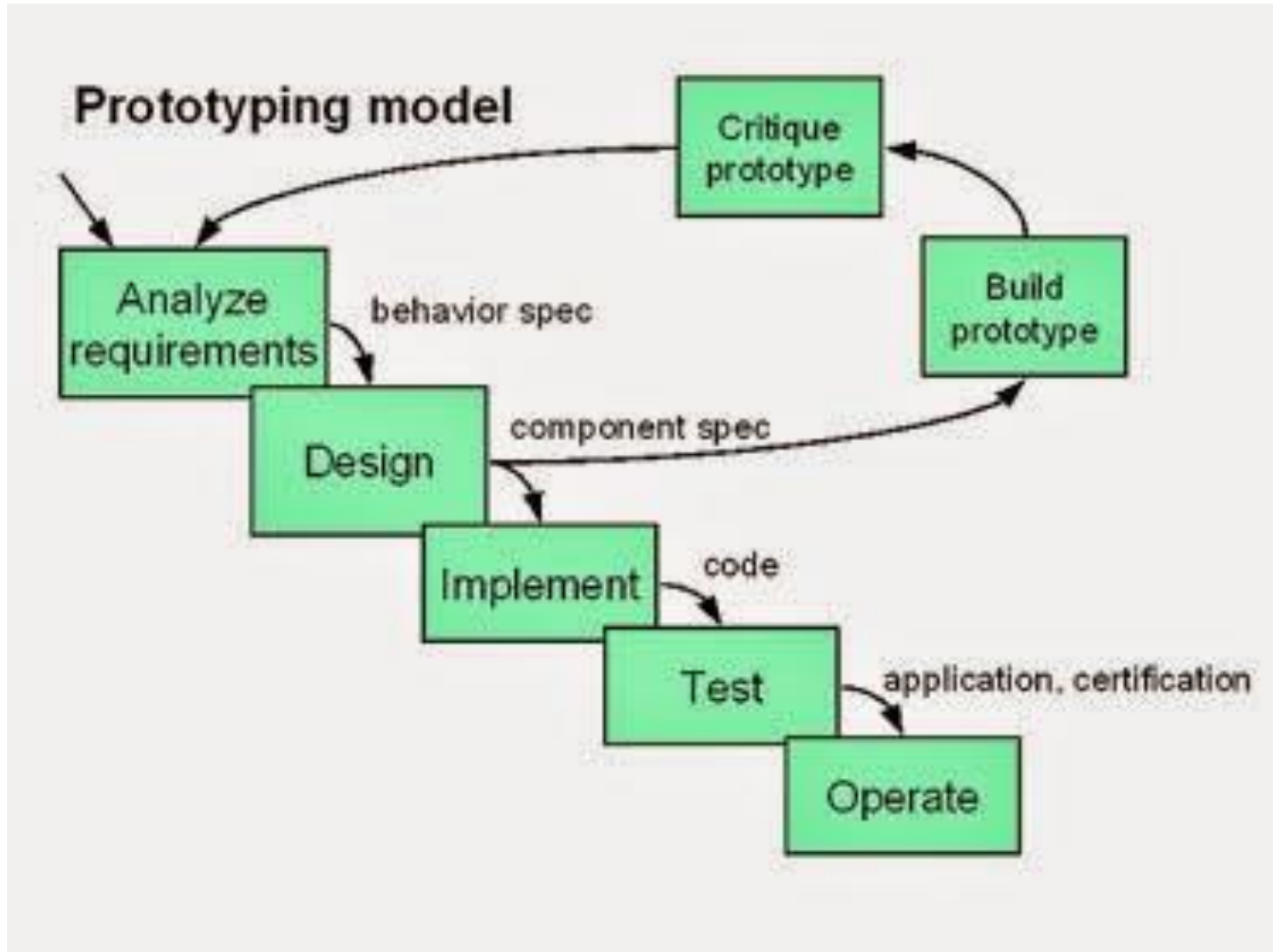
- Requirements are very clear.
- Quality is over speed.
- Manufacturing and construction projects that are similar to previous ones.
- Major project constraints are well understood and documented.



Prototyping

- It refers to the activity of creating prototypes of software applications, for example, incomplete versions of the software program being developed.
- It is an activity that can occur in software development and is used to visualize some component of the software to limit the gap of misunderstanding the customer requirements by the development team.
- This also will reduce the iterations may occur in the waterfall approach and hard to be implemented due to the inflexibility of the waterfall approach. So, when the final prototype is developed, the requirement is considered to be frozen.

Prototyping





Prototyping

Advantages

- This model is flexible in design.
- It is easy to detect errors.
- It can actively involve users in the development phase.
- It helps developers and users both understand the system better.
- We can find missing functionality easily.
- There is scope of refinement, it means new requirements can be easily accommodated.
- It can be reused by the developer for more complicated projects in the future.
- It ensures a greater level of customer satisfaction and comfort.
- Integration requirements are very well understood and deployment channels are decided at a very early stage.



Prototyping

Disadvantages

- This model is costly.
- It has poor documentation because of continuously changing customer requirements.
- There may be too much variation in requirements.
- Customers sometimes demand the actual product to be delivered soon after seeing an early prototype.
- There may be sub-optimal solutions because of developers in a hurry to build prototypes.
- Customers may not be satisfied or interested in the product after seeing the initial prototype.
- There may be incomplete or inadequate problem analysis.
- There may increase the complexity of the system.



Prototyping

When to Choose?

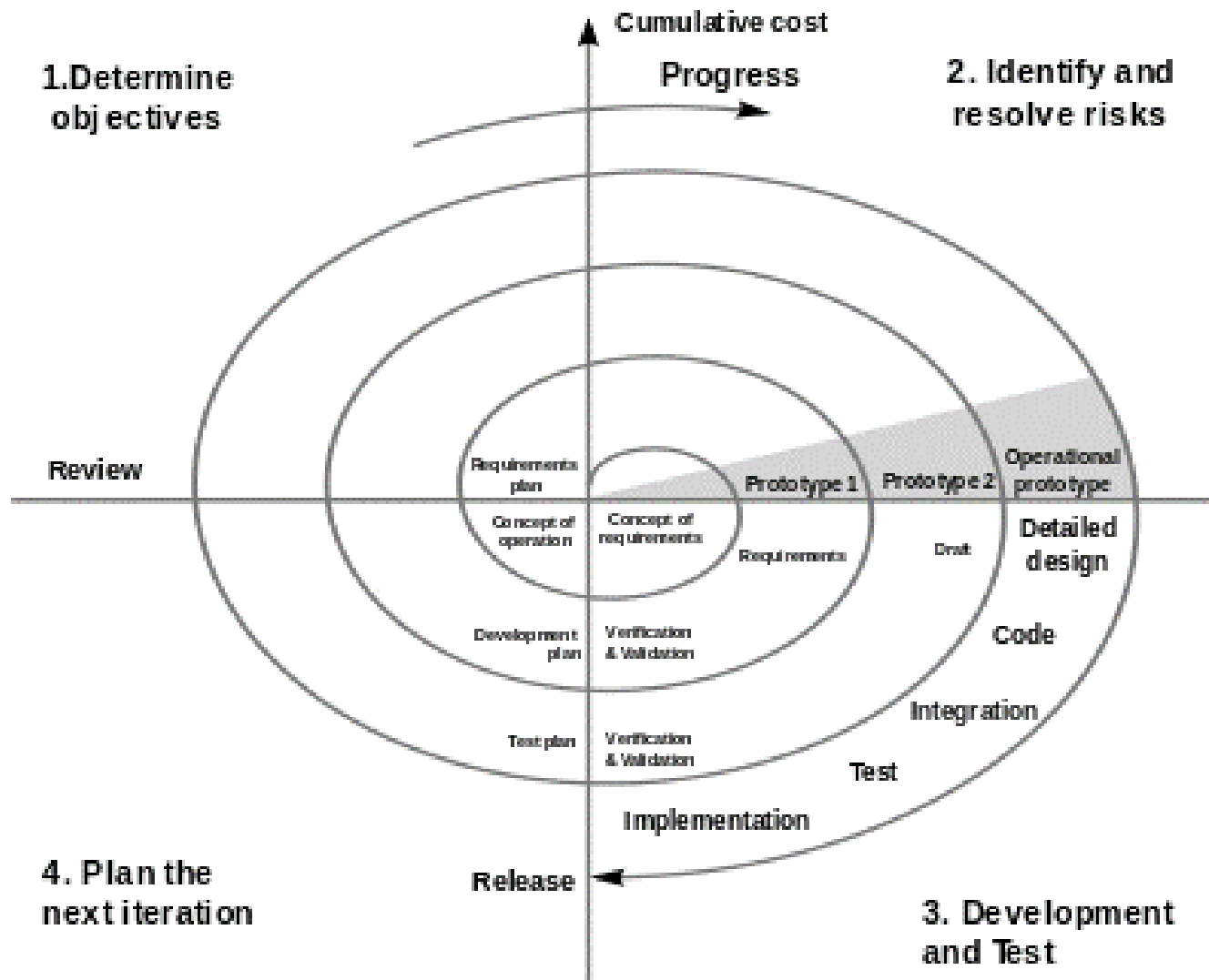
- This process can be used with any software developing life cycle model. But...
- **Prototype model should be used when the desired system needs to have a lot of interaction with the end users.**
- **Typically, online systems, web interfaces have a very high amount of interaction with end users, are best suited for prototype model.**



Spiral Model

- This model of development combines the features of the prototyping model and the waterfall model.
- The spiral model is favored for large, expensive, and complicated projects.
- This model uses many of the same phases as the waterfall model, in essentially the same order, separated by planning, risk assessment, and the building of prototypes and simulations.

Spiral Model





Spiral Model

Advantages

- **Software is produced early in the software life cycle.**
- **Flexibility in requirements.** In this model, we can easily change requirements at later phases and can be incorporated accurately. Also, additional functionality can be added at a later date.
- **It is good for large and complex projects.**
- **It is good for customer satisfaction.** We can involve customers in the development of products at early phase of the software development. Also, software is produced early in the software life cycle.
- **Strong approval and documentation control.**
- **Risk handling is one of important advantages.** It is best development model to follow due to the risk analysis and risk handling at every phase.
- **It is suitable for high risk projects, where business needs may be unstable.** A highly customized product can be developed using this.



Spiral Model

Disadvantages

- It is not suitable for small projects as it is expensive.
- It is much more complex than other SDLC models.
- Too much dependable on risk analysis and requires highly specific expertise.
- Difficulty in time management. As the number of phases is unknown at the start of the project, so time estimation is very difficult.
- Spiral may go on indefinitely.
- End of the project may not be known early.
- It is not suitable for low risk projects.
- May be hard to define objective, verifiable milestones. Large numbers of intermediate stages require excessive documentation.



Spiral Model

When to Choose?

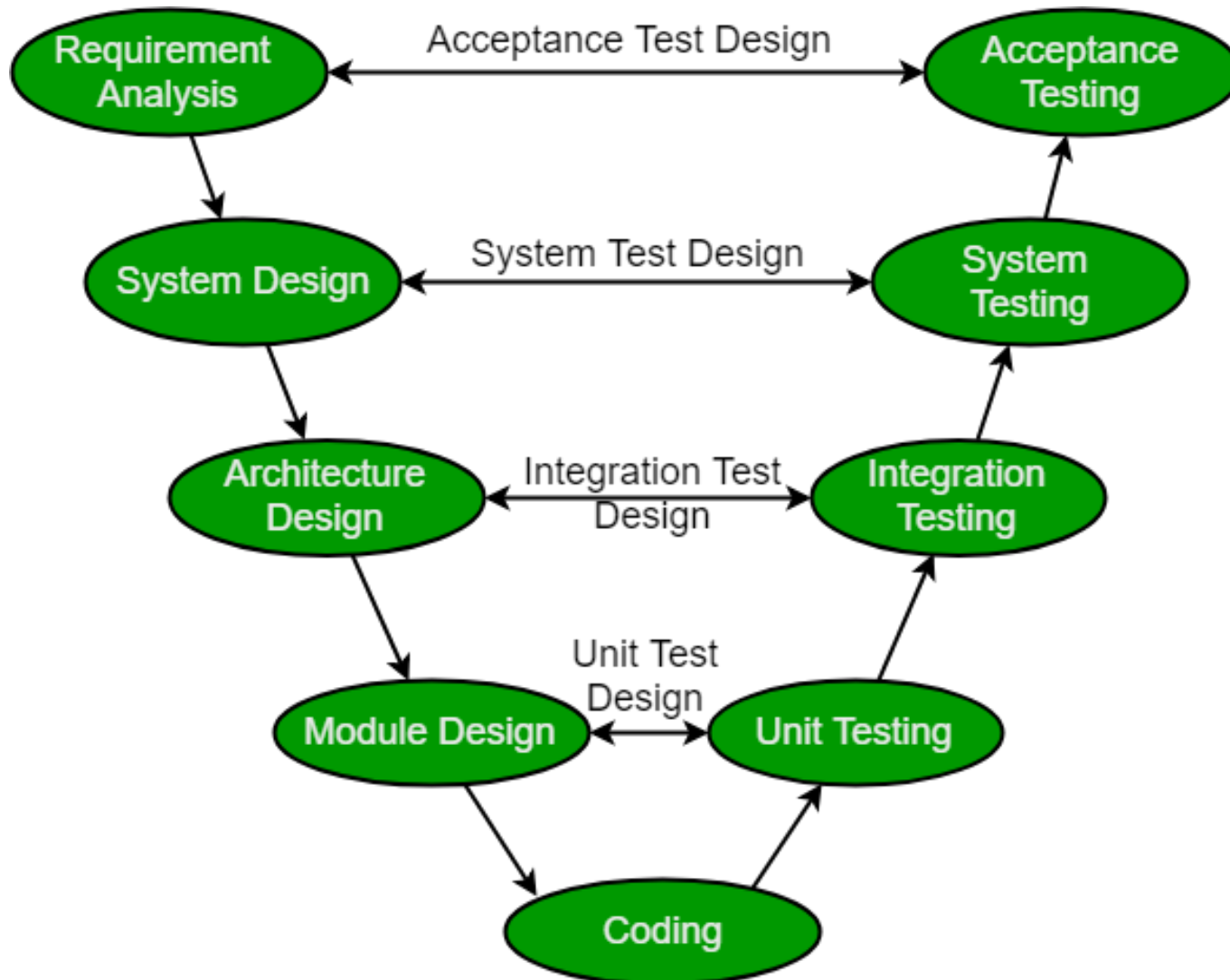
- **A spiral model is used when project is large.**
- **When releases are required to be frequent, spiral methodology is used.**
- **When creation of a prototype is applicable.**
- **When risk and cost evaluation is important.**
- **Spiral methodology is useful for medium to high-risk projects.**
- **When requirements are unclear and complex.**
- **When changes may require at any time.**
- **When long term project commitment is not feasible due to changes in economic priorities.**



V-Shaped Model

- It is an extension of the waterfall model.
- Instead of moving down in a linear way, the process steps are bent upwards after the implementation and coding phase, to form the typical V shape.
- The major difference between the V-shaped model and waterfall model is the early and more structural test planning in the V-shaped model.

V-Shaped Model





V-Shaped Model

Advantages

- Simple and easy to use.
- Time saving (because of simplicity and dedicated test phases).
- Each phase has specific deliverables.
- Higher chance of success over the waterfall model due to the development of test plans early on during the life cycle.
- Verification and validation of the product in the early stages of product development.



V-Shaped Model

Disadvantages

- Very inflexible, like the waterfall model.
- Adjusting scope is difficult and expensive.
- The software is developed during the implementation phase, so no early prototypes of the software are produced.
- The model doesn't provide a clear path for problems found during testing phases.



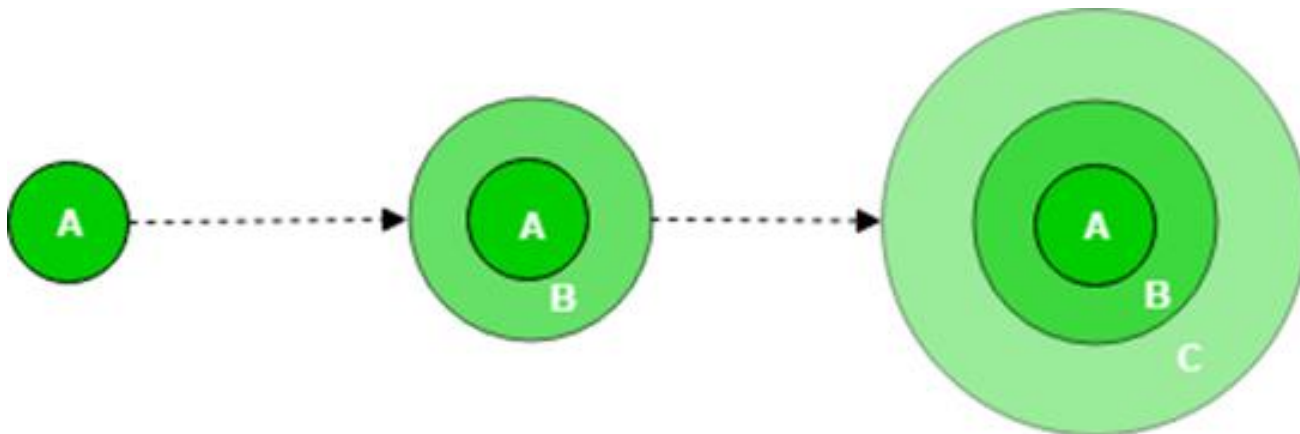
V-Shaped Model

When to Choose?

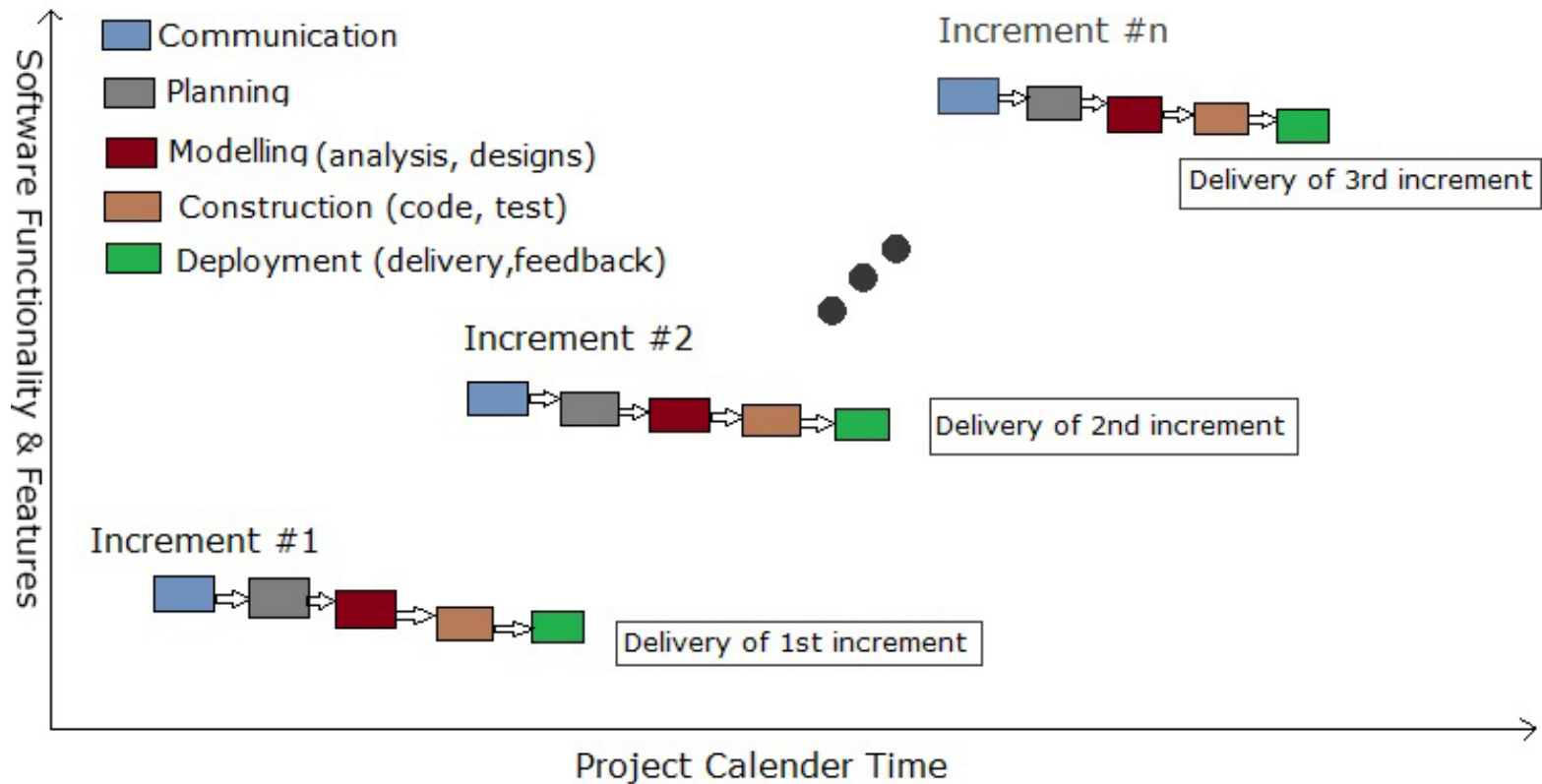
- Works well for where requirements are clearly defined and known.

Incremental Development

- Requirements are broken down into multiple standalone modules of software development cycle.
- Incremental process model is also known as Successive Version Model.
- First, a simple working system implementing only a few basic features is built and then that is delivered to the customer. Then thereafter many successive iterations/ versions are implemented and delivered to the customer until the desired system is released.



Incremental Development



- The first increment is often a core product where the basic requirements are addressed, and supplementary features are added in the next increments.



Incremental Development

Advantages

- The software will be generated quickly during the software life cycle.
- It is flexible and less expensive to change requirements and scope.
- A customer can respond to each building.
- Error reduction (core modules are used by the customer from the beginning of the phase and then these are tested thoroughly).
- Uses divide and conquer for breakdown of tasks.
- Lowers initial delivery cost. This model is less costly compared to others.
- Incremental resource deployment.



Incremental Development

Disadvantages

- Requires good planning and design.
- Problems might cause due to system architecture as such not all requirements collected up front for the entire software lifecycle.
- Each iteration phase is rigid and does not overlap each other.
- Well defined module interfaces are required.
- Rectifying a problem in one unit requires correction in all the units and consumes a lot of time.



Incremental Development

When to Choose?

- **When projects having lengthy developments schedules.**
- **Projects with new technology.**
- **Requirements of the system are clearly understood.**
- **When demand for an early release of a product arises.**
- **When software engineering team are NOT very well skilled or trained.**
- **When high-risk features and goals are involved.**
- **Such methodology is more in use for web application and product based companies.**



Prototyping vs Incremental Development?

- **Prototyping:** Instead of freezing the requirements before a design or coding can proceed, a throwaway prototype is built to understand the requirements. This prototype is developed based on the currently known requirements.
- A prototype can be partially complete to explore some aspects of the system requirements.
- Prototyping is not necessarily concerned with delivering a working system whereas incremental model delivers a working system in successive increments.



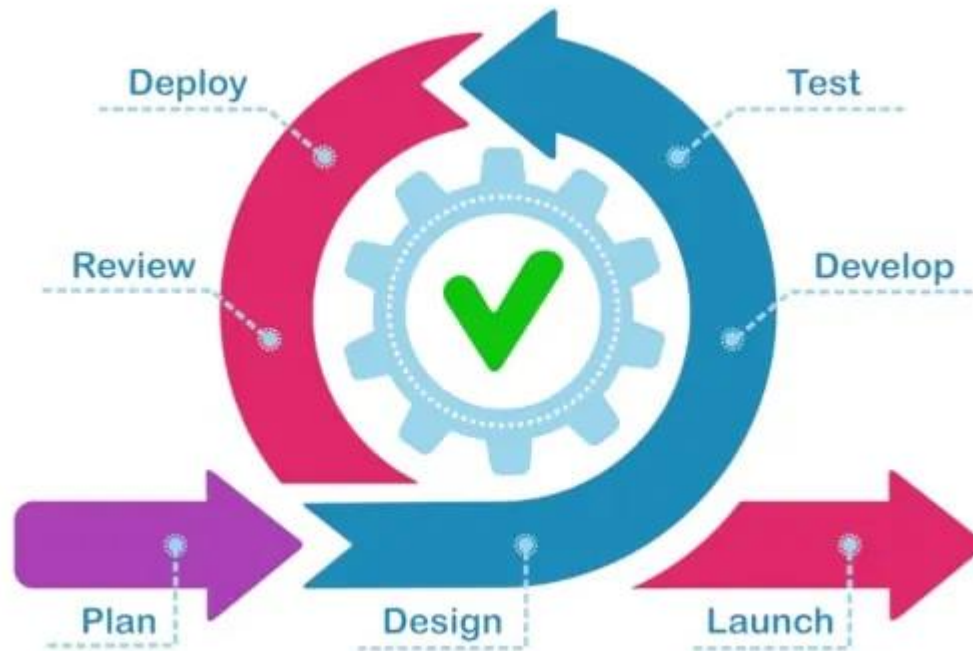
The Agile Family

- **To overcome drawbacks of Waterfall model, in the mid-1990s the Agile Software Development model was proposed. The core of the Agile methodology was developed in 2001 in written form.**
- **Agile Manifesto of Software Development put forth a groundbreaking mindset on delivering value and collaborating with customers.**
- **Agile methods ensure:**
 - **Constant and continuous communication.**
 - **Customer collaboration.**
 - **Team collaboration.**
 - **Response to changes.**
 - **Readiness of a working product.**

The Agile Family

- Scrum
- Kanban
- Lean Development
- Extreme Programming (XP)
- Adaptive Software Development (ASD)
- Feature Driven Development (FDD)
- Dynamic Systems Development Method (DSDM)
- Crystal

The Agile Family





The Agile Family

Advantages

- Early and frequent releases.
- Accommodation of changing requirements.
- Daily communication among the customer and developers.
- Reduce unnecessary documentation.
- Self-organizing teams.
- Simplicity, focusing on what is immediately required.
- Regular reflection to adjust behavior to improve effectiveness.



The Agile Family

Disadvantages

- Customer availability may not be possible.
- Teams should be experienced to follow the rules of the method.
- Appropriate planning is required to quickly decide on the functionality that needs to be delivered in an iteration.
- Team is expected to have estimation skills and negotiation skills.
- Team should have effective communication skills.
- Difficult to develop and deliver in time-boxed iterations.



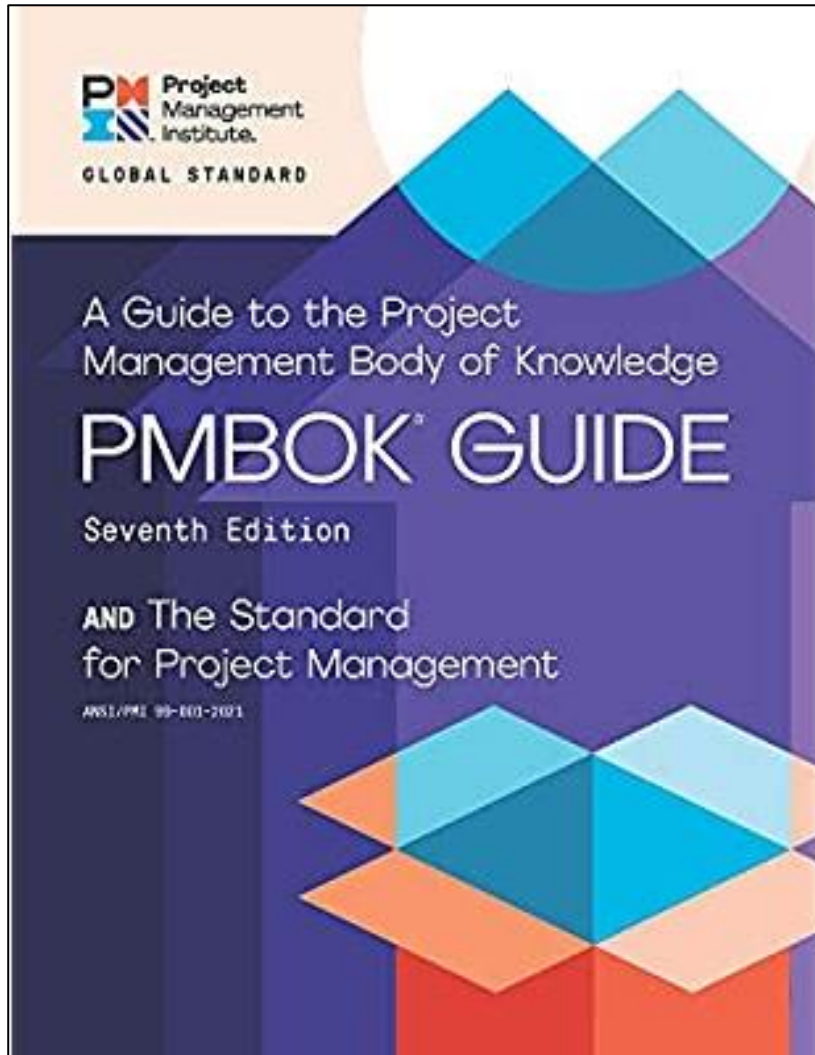
The Agile Family

When to Choose?

- **Application is time-critical.**
- **Organization employs disciplined methods.**
- **Unclear requirements/deliverables.**
- **Level of participation/input and buy-in from stakeholders.**
- **Cost of change is minimal.**
- **Emphasis on teamwork, transparency and continuous improvement.**



You Should Know These Two!



You Should Know These Two!

Project Management Body of Knowledge (PMBOK)

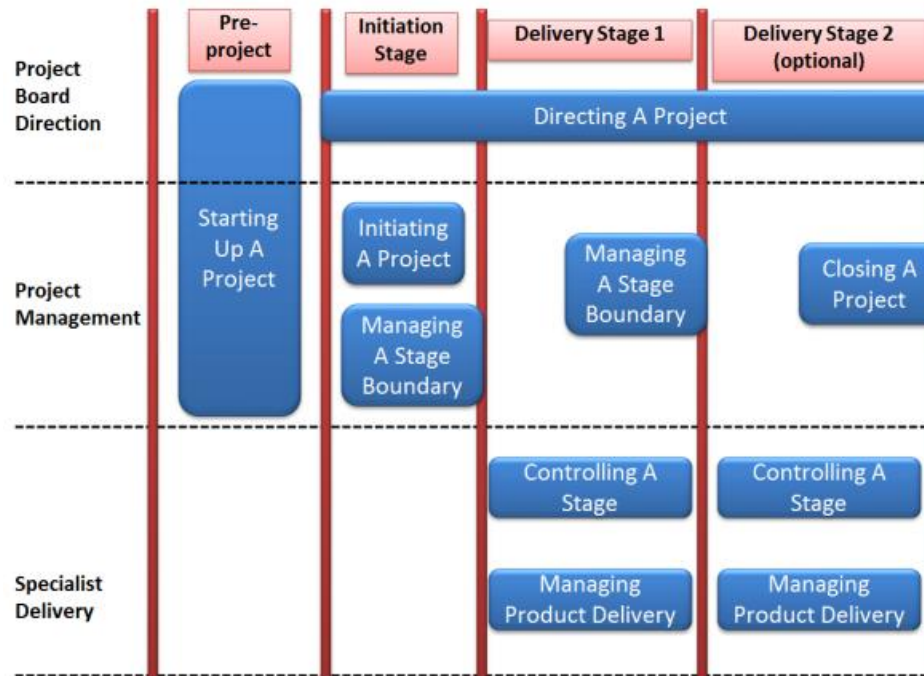
- **PMBOK is a framework of standards, conventions, processes, best practices, terminologies, and guidelines that are accepted as project management industry standards.**
- **Process-oriented.**
- **It defines for each process the necessary input, tools, techniques and output (deliverables).**



You Should Know These Two!

Projects IN Controlled Environments (PRINCE2)

- PRINCE2 is the official project management methodology of the UK government.
- Not to be confused with the PMBOK (a best-practices resource), PRINCE2 is a complete project methodology system.
- This methodology is best suited for large and complex projects with fixed requirements.





You Should Know These Two!

PMBOK vs PRINCE2

PMBOK	PRINCE2
Framework	Methodology
Activity-focused	Delivery-focused
Customer requirements driven	Business case driven
5 process groups 10 knowledge areas	7 principles 7 themes
49 processes	7 processes



SDLC Workshop

Project 1

- Web-based software, integrated with multiple dispersed subsystems.
- There may be a need for novel technological solutions.
- You have a team with a mixture of experienced and novice guys.
- Choose **Spiral or Incremental**.

Project 2

- Sending humans to the Moon.
- You are the Turkish government. So, imagine the resources of Turkey when considering methodology.
- Choose **Prototyping or Incremental**.

Project 3

- University course management system like UBIS.
- You are a small software company.
- Choose **Waterfall or Incremental**.

Project 4

- An autonomous car.
- You are a middle-sized company in Istanbul, with about 100 engineers from different technology domains.
- You collaborate with South Korea.
- Choose **Waterfall or V-Shape**.

Directions

- Consider and list all of the factors for choosing methodology.
- List all the factors that prevented you from selecting the other methodology.
- Write down all of your assumptions, including time, budget, stakeholders etc.

Which Methodology?



1



2



3



4



1



2



3

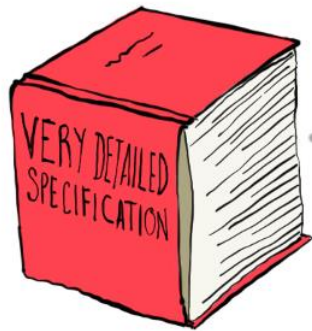


4



5

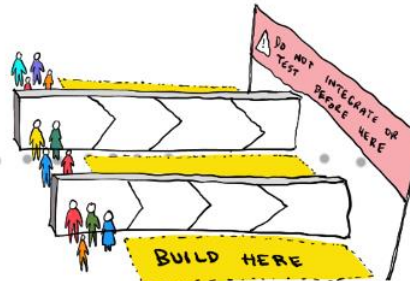
Which Methodology?



**Gather
Requirements**



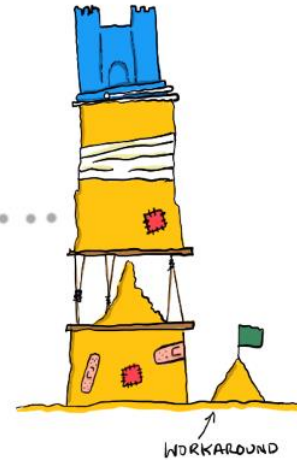
Design



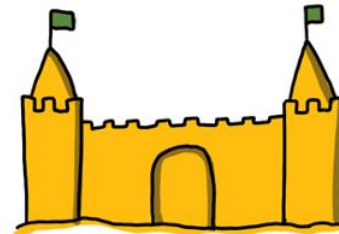
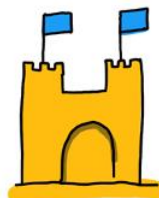
Build and Test



Err...



Deployment





Thank you...

Hasan ÇİFCİ