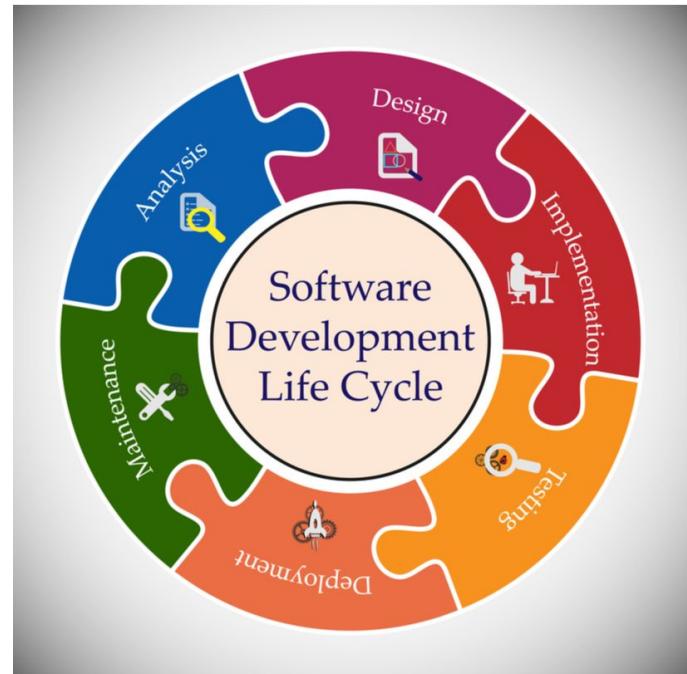




SDLC Session - 1



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

SDLC



Software Development Life Cycle

CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Circle how you are feeling:



Pear Deck



Students, draw anywhere on this slide!

Pear Deck Interactive Slide
Do not remove this bar

XYZ TABLE AND CHAIR COMPANY



Table Production Life Cycle (TPLC)



BEFORE (in-class session)



What do you know about SDLC.

(Please write shortly on PEAR DECK slide)



Students, write your response!

Pear Deck Interactive Slide
Do not remove this bar

5





Table of Contents

- ▶ What is SDLC ?
- ▶ Phases of SDLC
- ▶ SDLC Models
- ▶ Waterfall Model

* New approaches (agile and devops) and their implementations (Jira) will be explained later as separate lessons.

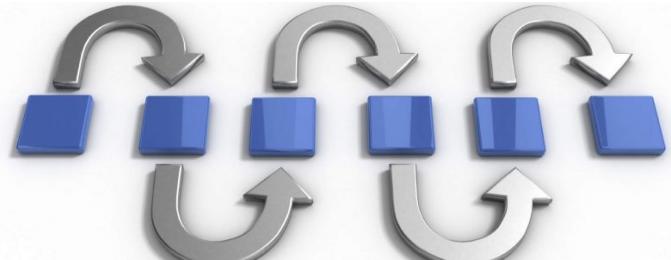
7



1

What is SDLC ?

What is SDLC



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

9

What is SDLC

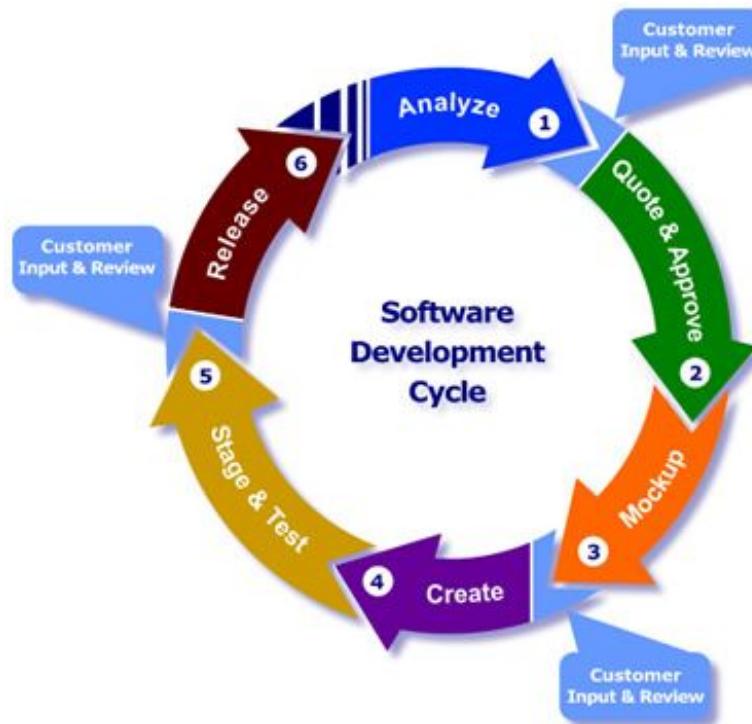


- Systematic process to be followed for a software project.
- Structured way to create and develop software.

CLARUSWAY[©]
WAY TO REINVENT YOURSELF

10

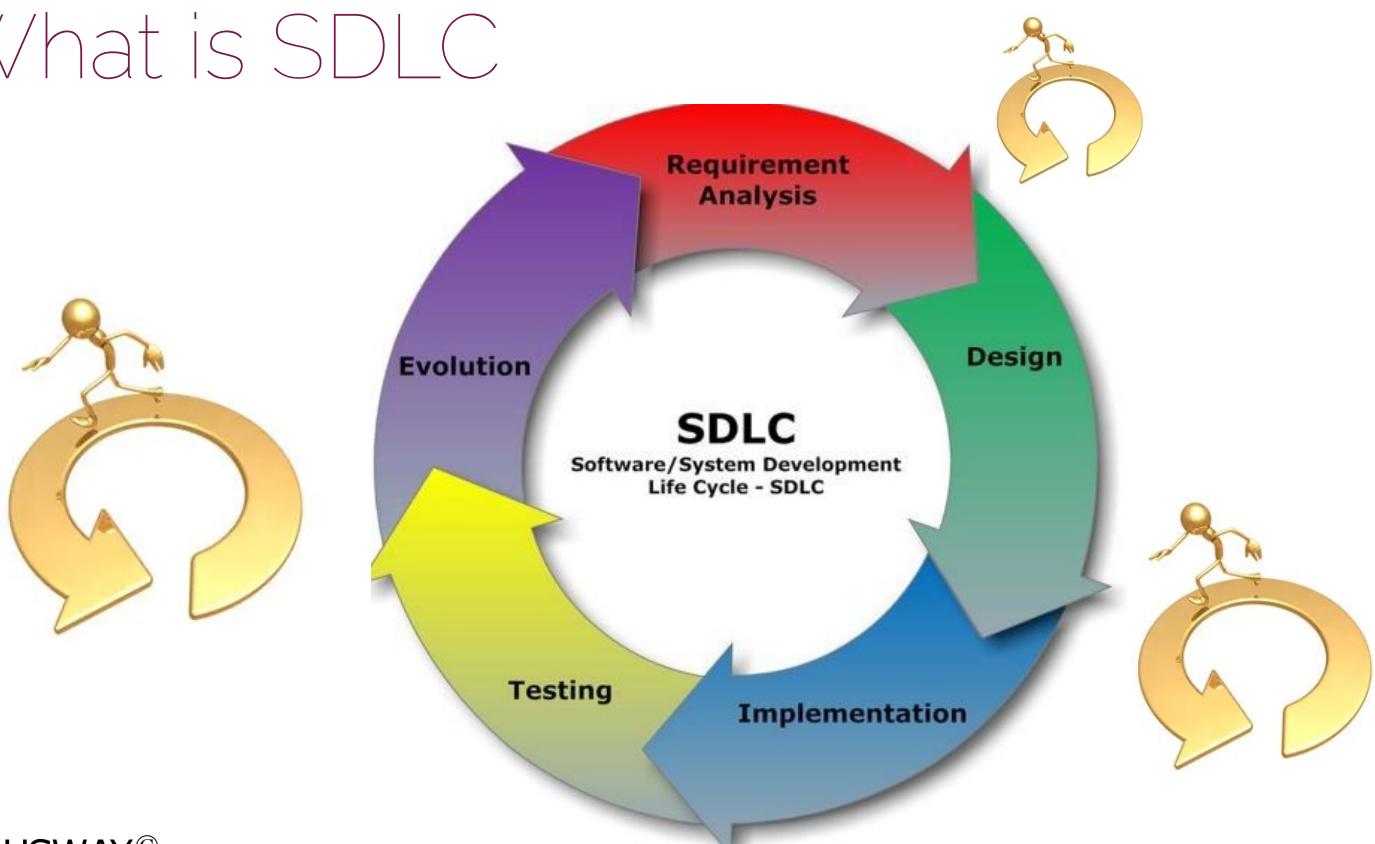
What is SDLC



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

11

What is SDLC



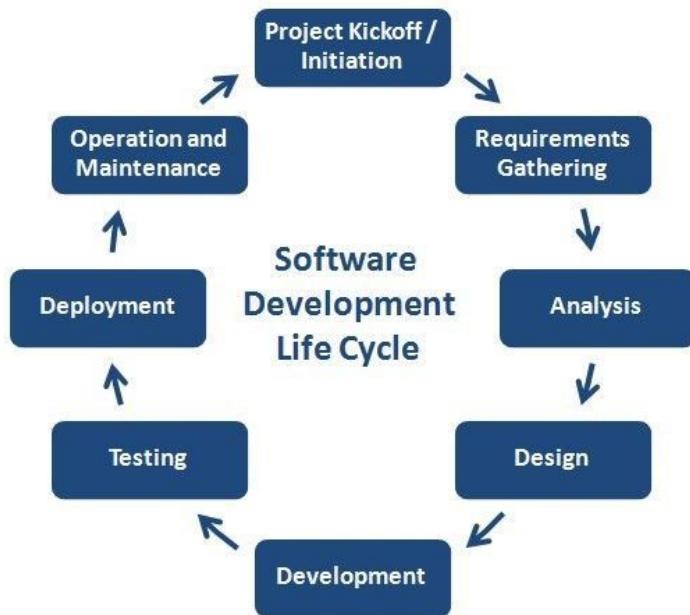
CLARUSWAY[©]
WAY TO REINVENT YOURSELF

12



What is SDLC

V1.02.03



13

CLARUSWAY[©]
WAY TO REINVENT YOURSELF



timtim.com

CLARUSWAY[©]
WAY TO REINVENT YOURSELF



2

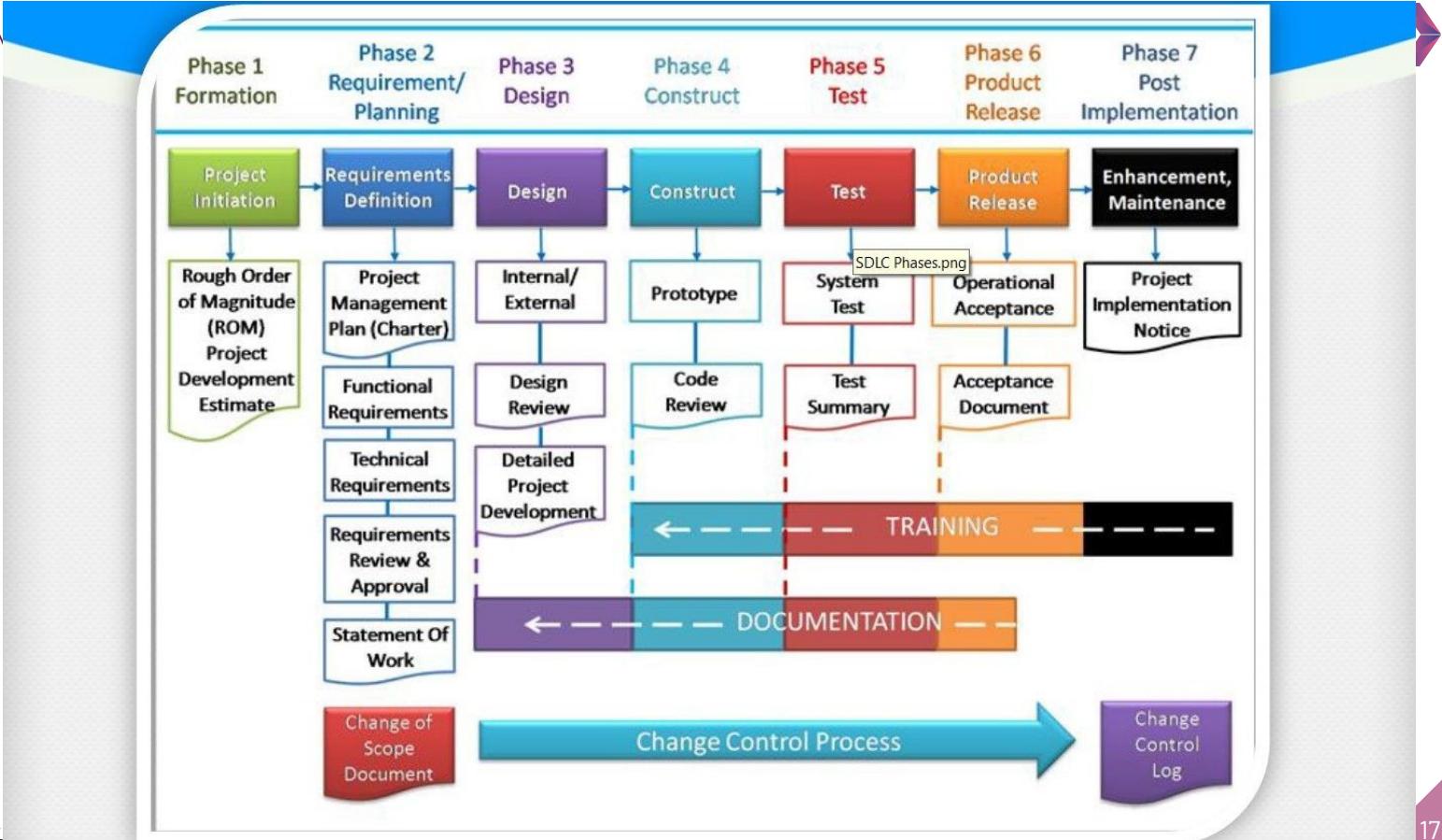
Phases of SDLC

CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Phases of SDLC

How many phases does SDLC have?

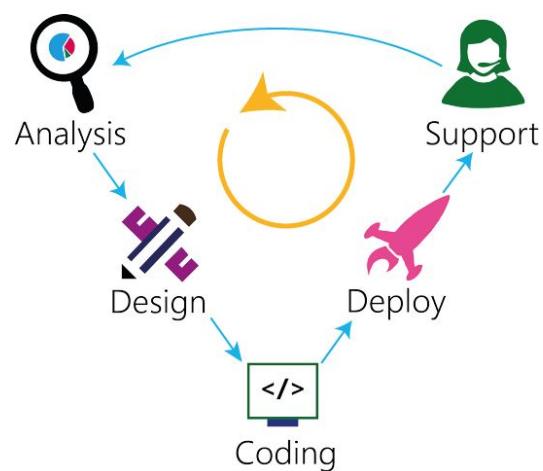




CLA
WAY TO RECOVER YOURSELF

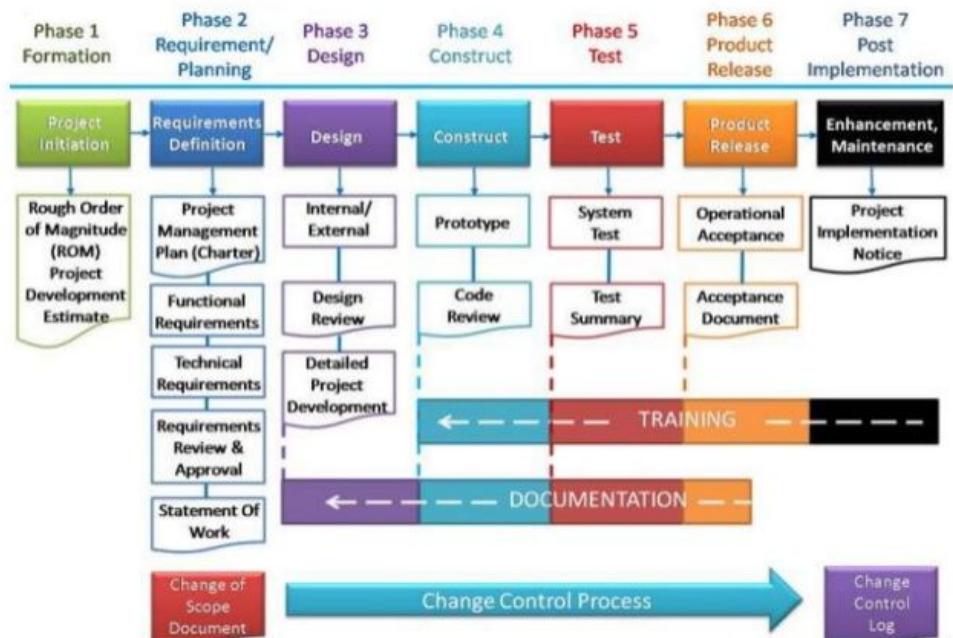
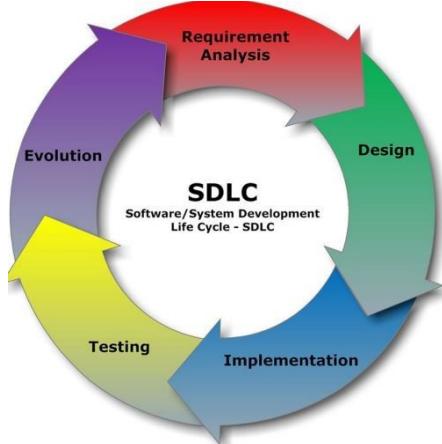
17

Phases of SDLC



18

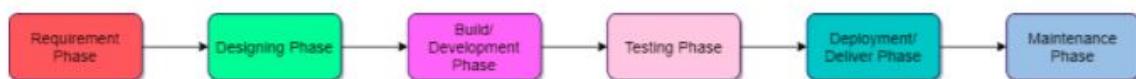
Phases of SDLC



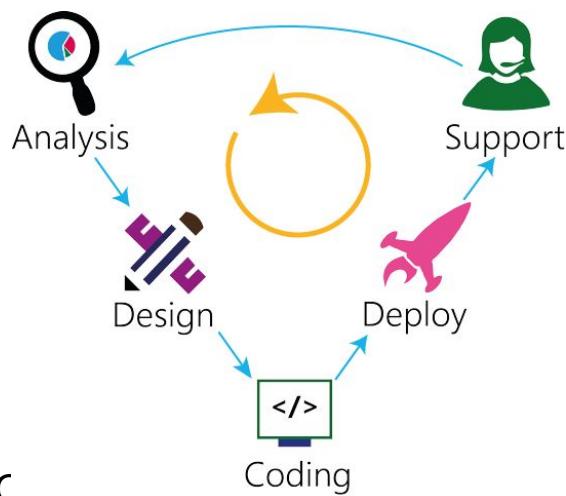
Phases of SDLC

The SDLC process consists essentially of the following phases:

- Requirement Phase
- Design Phase
- Build/Development Phase
- Testing Phase
- Deployment/Deliver Phase
- Maintenance



Phases of SDLC



21

Phases of SDLC

What is the most critical phase?





► Phases of SDLC

What is the name of the document that consists of all necessary requirements to be designed?



Students choose an option

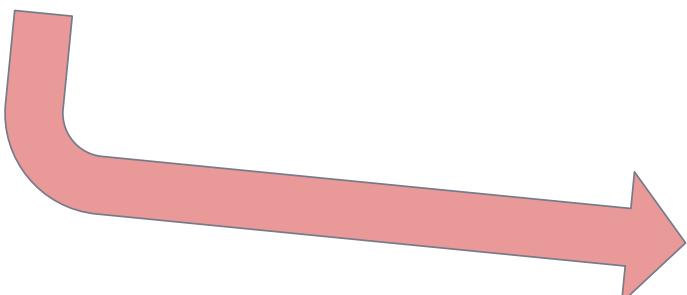
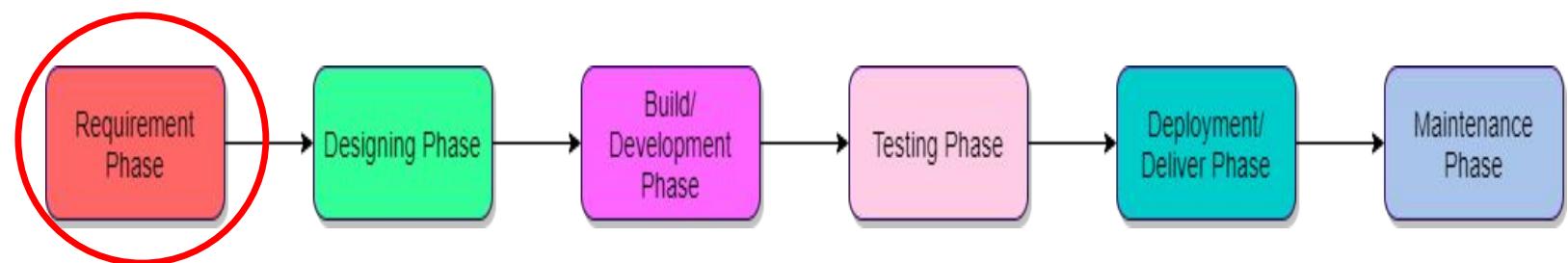
REINVENT YOURSELF

Table of Contents	
Introduction:	4
1.1 Purpose:	4
1.2 Document Conventions	4
1.3 Project Scope	5
1.3.1 Sales Manager	5
1.3.2 Inventory Manager	5
1.3.3 Customer Scope	5
1.4 References	5
2. Overall Descriptions	6
2.1 Product Perspective	6
2.1.1 Context Diagram	7
2.2 User Classes	8
2.3 Other Descriptions	9

Pear Deck Interactive Slide
Do not remove this bar

23

► Requirements Phase



Requirements Phase



Requirement Analysis

Design

Build/
Development

Testing

Deploy
Deliver

Maintena
nce



Requirement Phase



STRUCTURE OF SRS

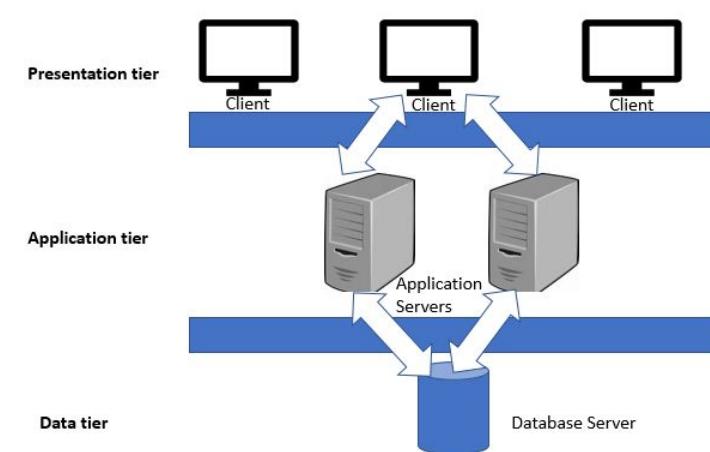
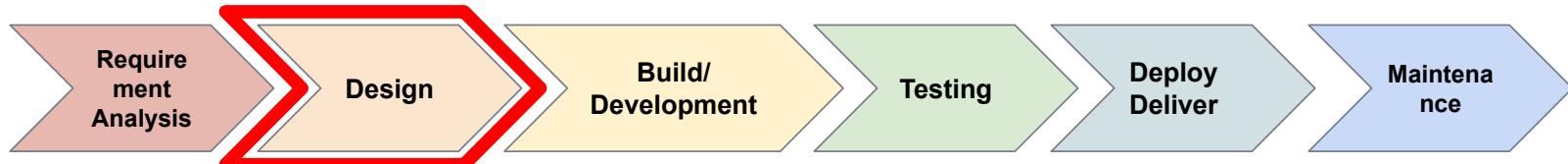
Chapter no. 1	Preface	It briefly explains about project.
Chapter no. 2	Introduction	Highlights the projects with its title and briefly describe the projects.
Chapter no. 3	Scope	What is the capability of the product?
Chapter no. 4	Glossary	Definition, acronyms and abbreviation.
Chapter no. 5	User requirement definition	Describes non-functional requirements
Chapter no. 6	Architecture	Specifies system architecture
Chapter no. 7	System requirements	System description with function and non-function requirement.
Chapter no. 8	System model	System model used to represent relationship.
Chapter no. 9	System evaluation	How system is evolved?
Chapter no. 10	Appendices	Annexure, application, data requirements.
Chapter no. 11	indexes	Indices of diagram, tables, functions.

Requirement Phase



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

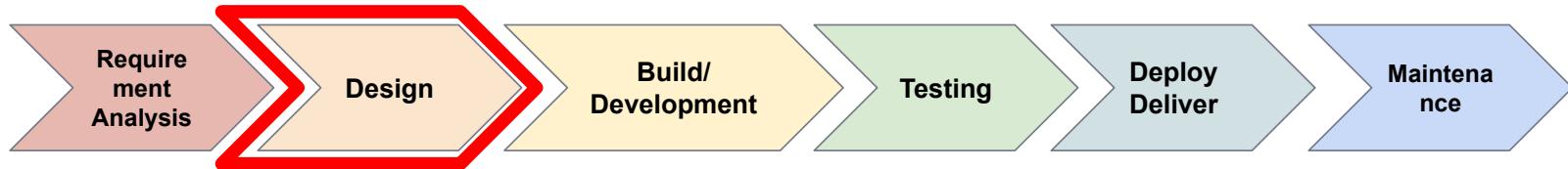
► Design Phase



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

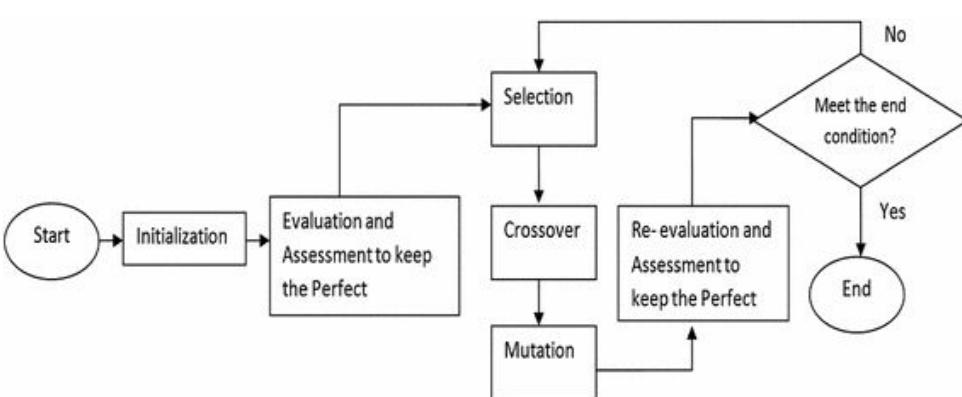
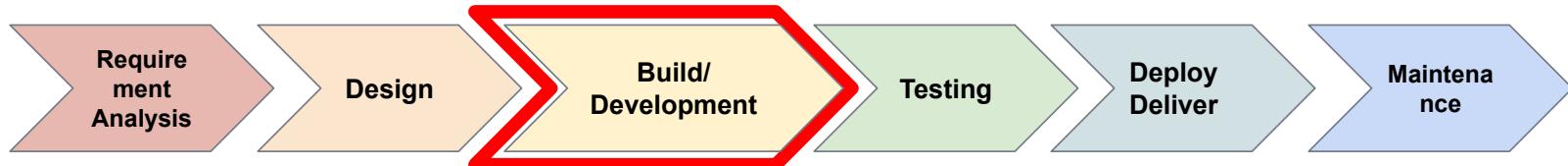


Design Phase



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Build/Development Phase

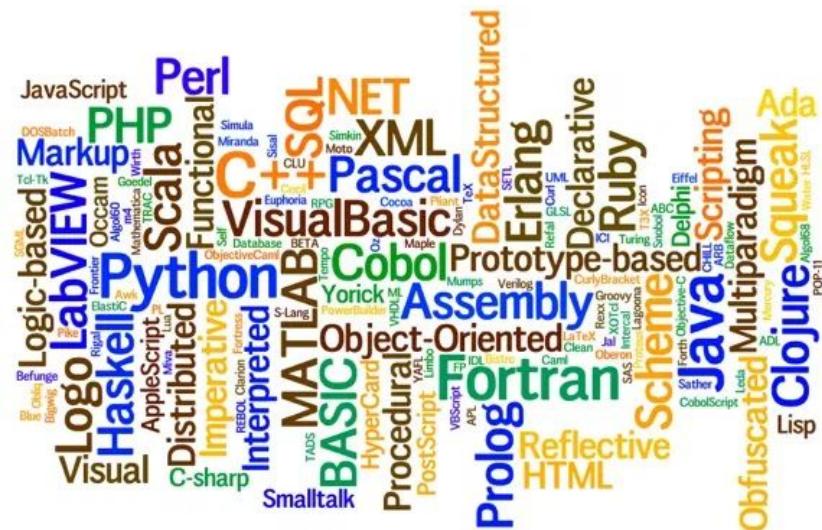
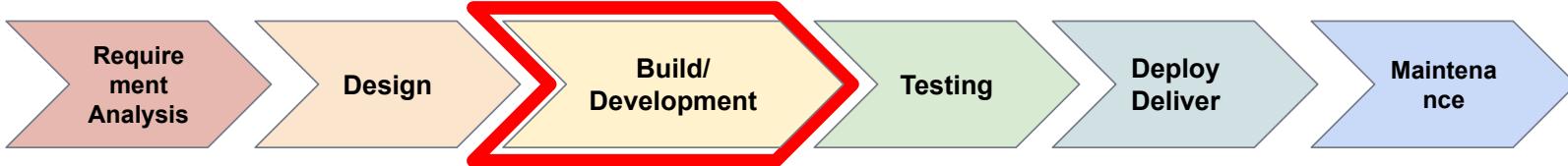


```

args.style; }) rest != 999) window.onload=do
  seconds();) args = arg1; </script> var str=span.
removeChild if(data.substring(i,i+1)==":") (sp
& res1 == fun(sp) {var theSpan=document.cre
(res1 = args.toString() document.createTextNode(
percent1++;window.status="% complete";
secForm = Math.floor(secTimeCode); sec.ctref
on Seconds(data) { :var ll = return(data.sub
r.whle(ll%4 != 0) var sd = name.value; bhspdr
360); else color.length=span.firstChild.data.le
(cube) { string.speed=(spd==fun(bar): if(isNur
= decimalToBin(sd); sqr.hinc=fork.deg/this.
  
```

CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Build/Development Phase



	SPI - SDI	IN
I2C Master CLK	OUT	
I2C Master Data	IN/OUT	
CAP	IN	
CAP	IN/OUT	
CAP	TTL	

31

Phases of SDLC

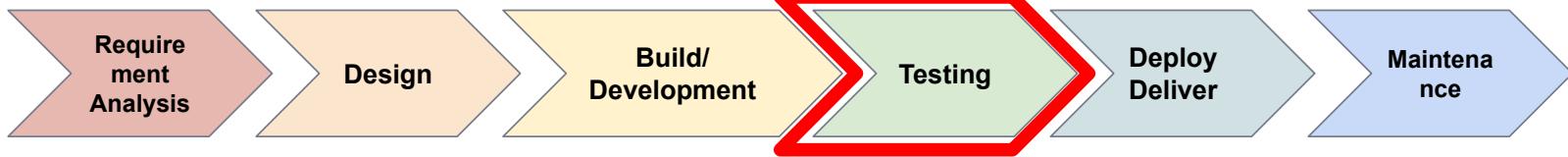


At what phase we focus on the investigation?





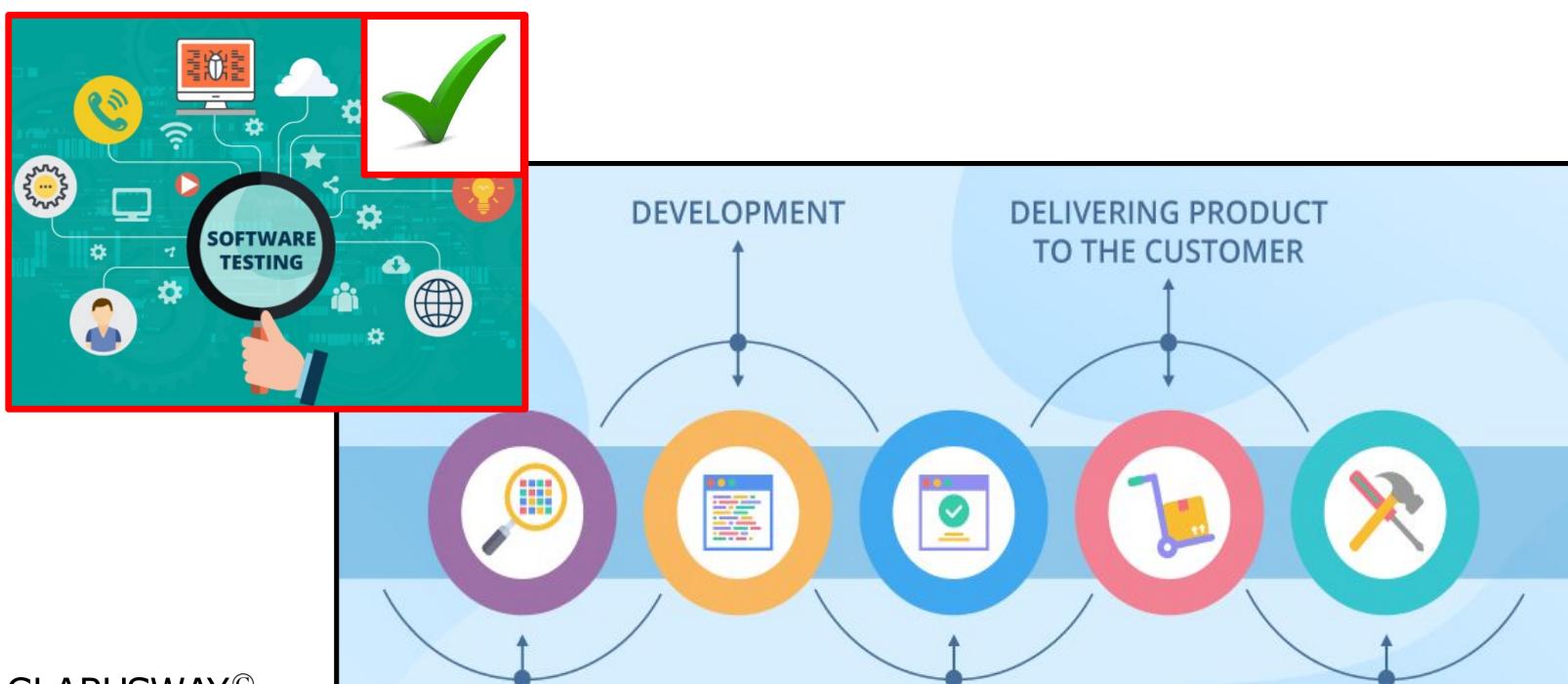
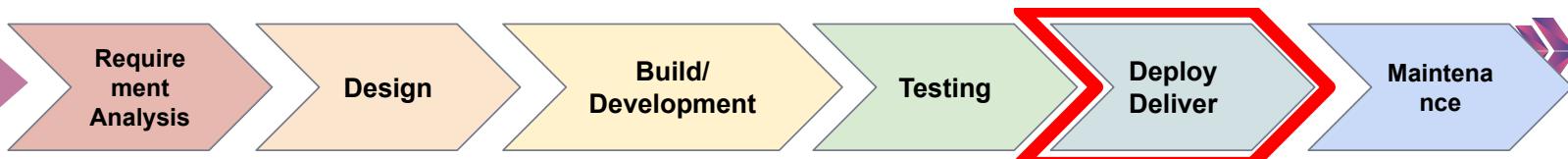
Testing Phase



CLARUSWAY[©]
WAY TO REINVENT YOURSELF



33



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

34

Requirement Analysis

Design

Build/
Development

Testing

Deploy
Deliver

Maintain-
ance



CLARUSWAY®
WAY TO REINVENT YOURSELF

Requirement Analysis

Design

Build/
Development

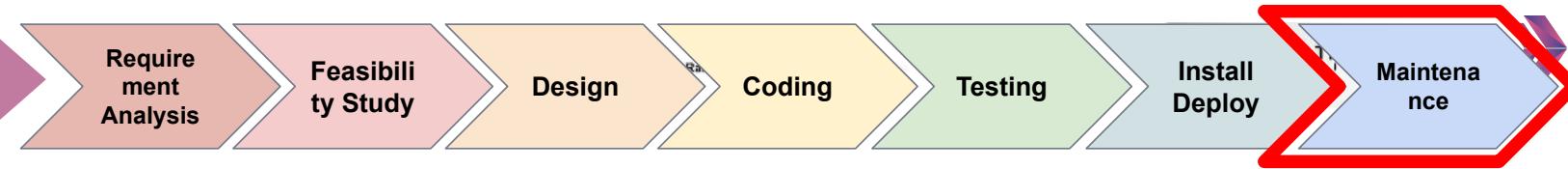
Testing

Deploy
Deliver

Maintain-
ance



CLARUSWAY®
WAY TO REINVENT YOURSELF



Verilerimizi gerçekten
şifrelemek zorunda mıyız?
Zaten başlangıçta
iletişimimizin büyük
bölümünü anlamak mümkün
değil ki...



**“Do we really need to encrypt our data? Most of
our communications are impossible to
understand in the first place.”**

SDLC Models



SDLC Models

List the common SDLC models.

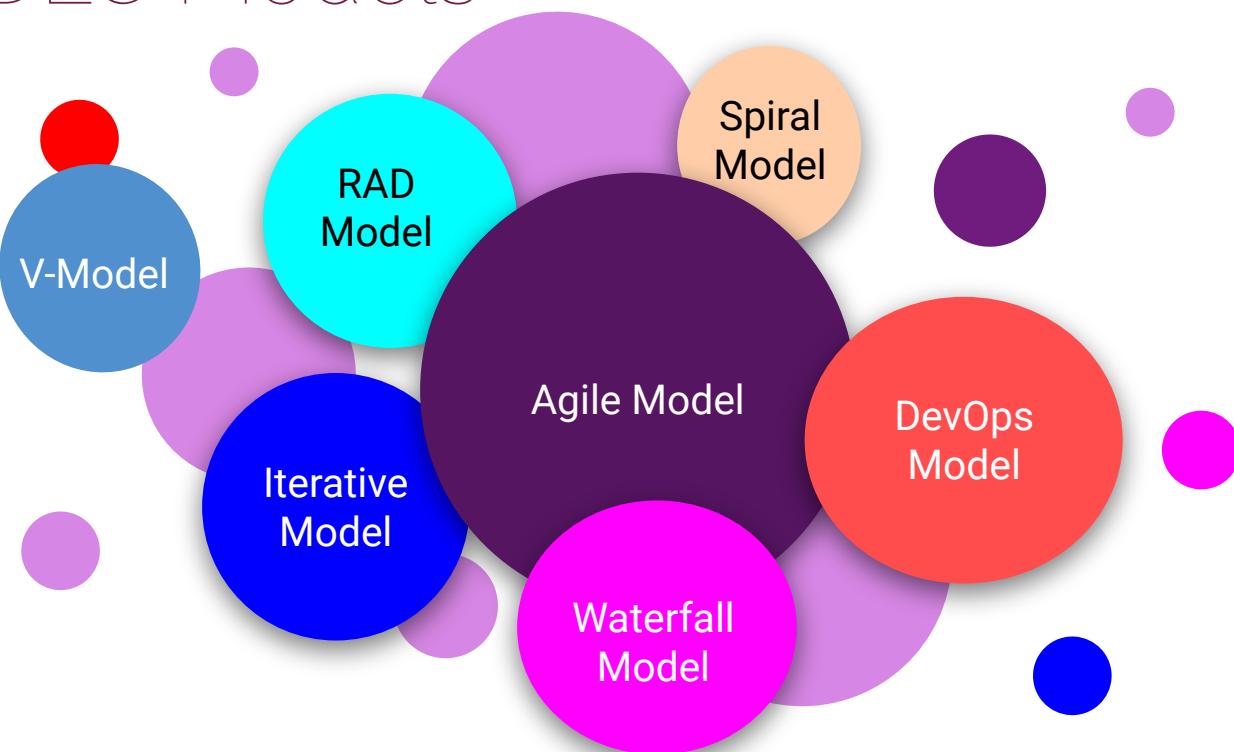


Students, write your response!

Pear Deck Interactive Slide
Do not remove this bar

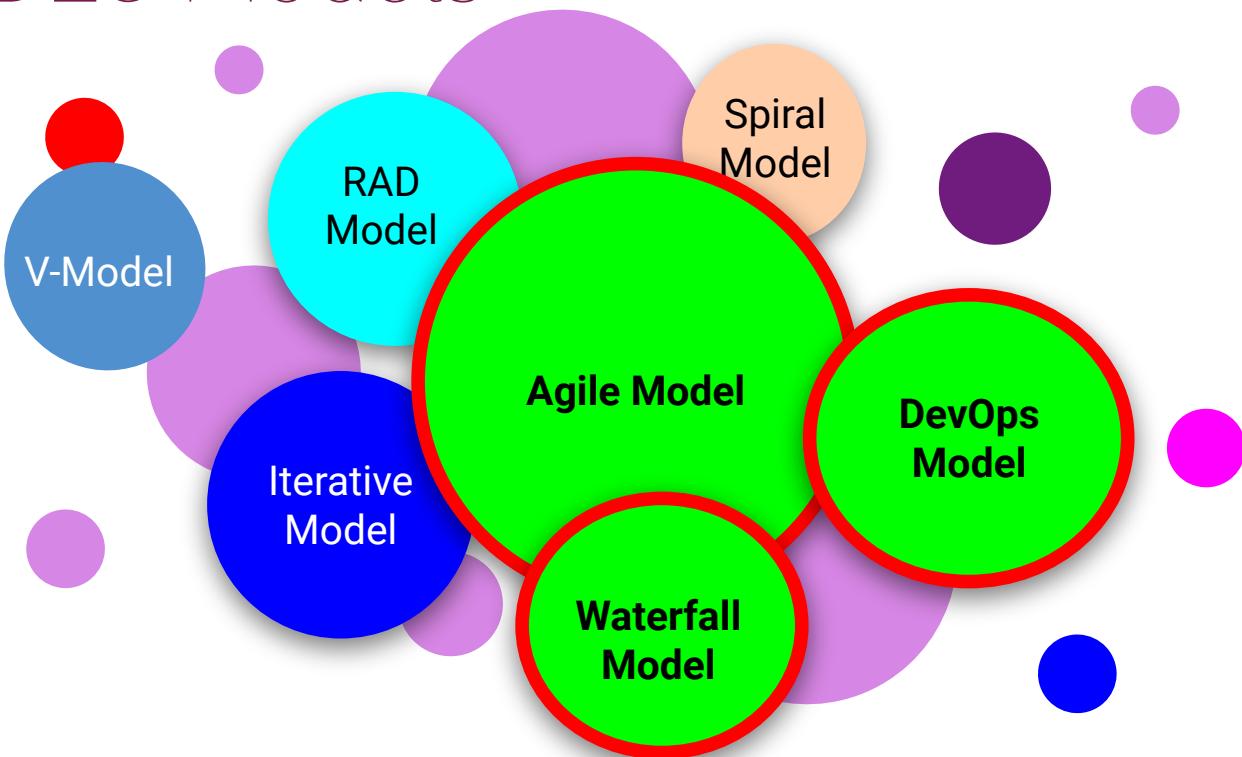
39

SDLC Models



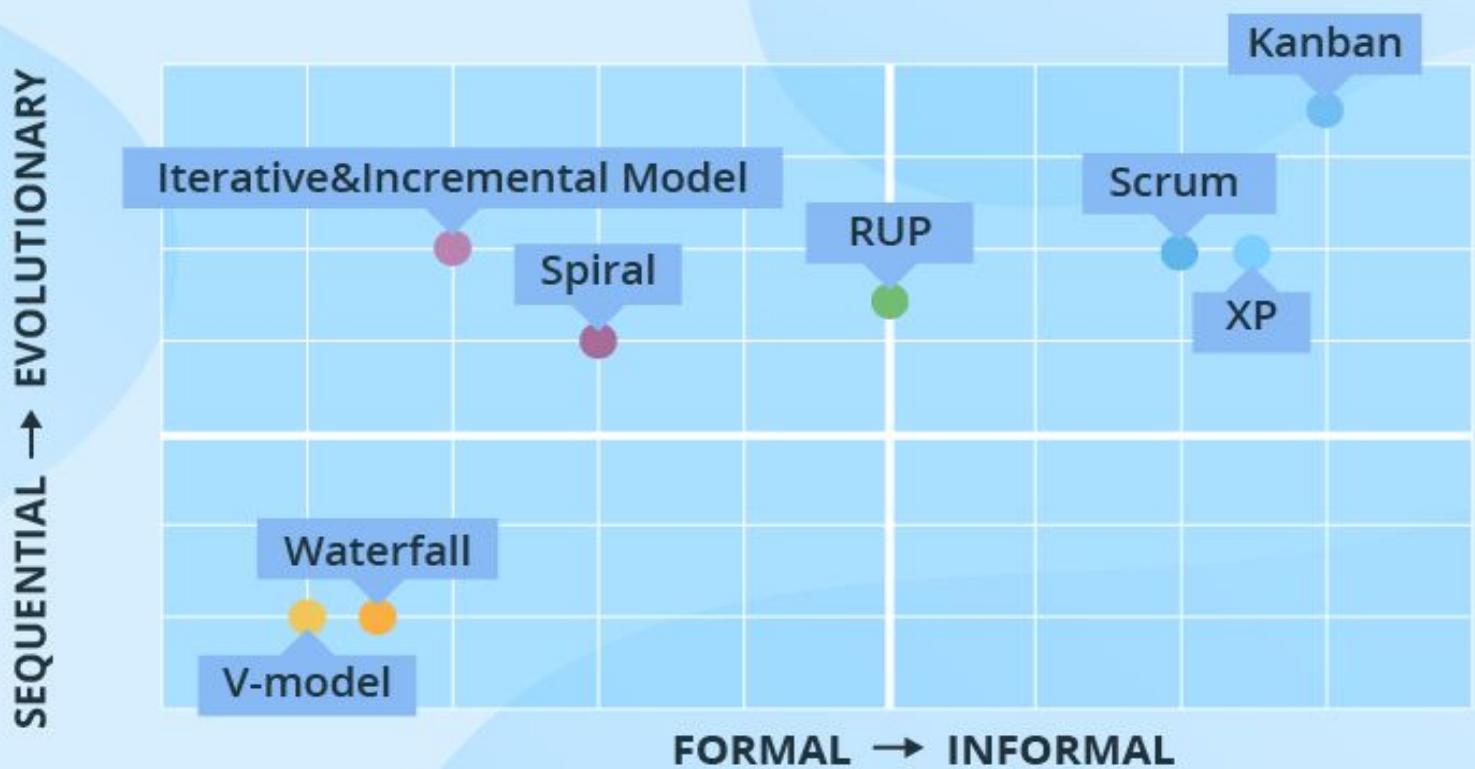


SDLC Models



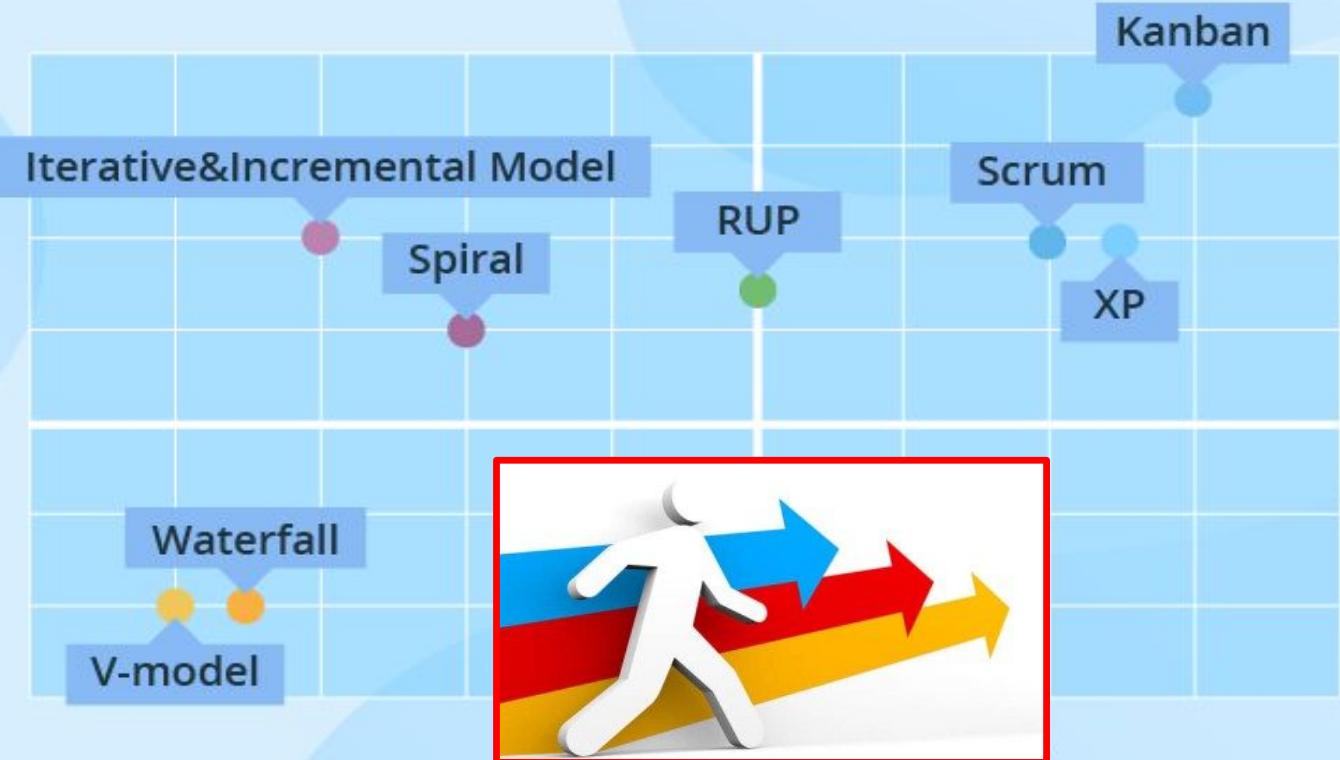
CLARUSWAY[©]
WAY TO REINVENT YOURSELF

TYPES OF POPULAR SDLC MODELS



TYPES OF POPULAR SDLC MODELS

SEQUENTIAL → EVOLUTIONARY



SDLC Models

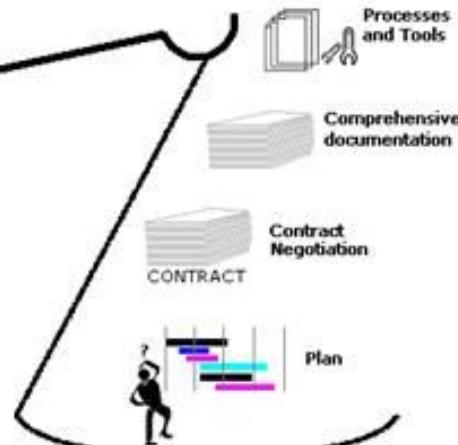
Which one is the traditional SDLC model?

SDLC Models

Agile Development



Traditional Development



CLARUSW,
WAY TO REINVENT YOURSELF



Waterfall Model

4



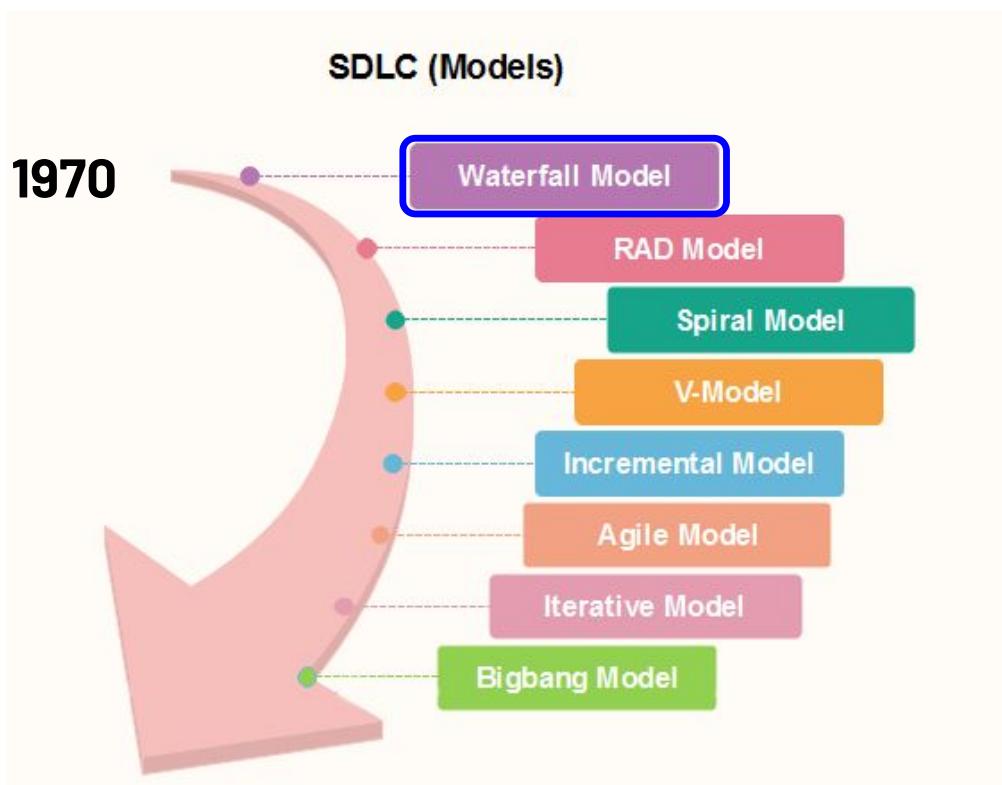
Waterfall Model

In which years did the Waterfall model appear?





Waterfall Model



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

49

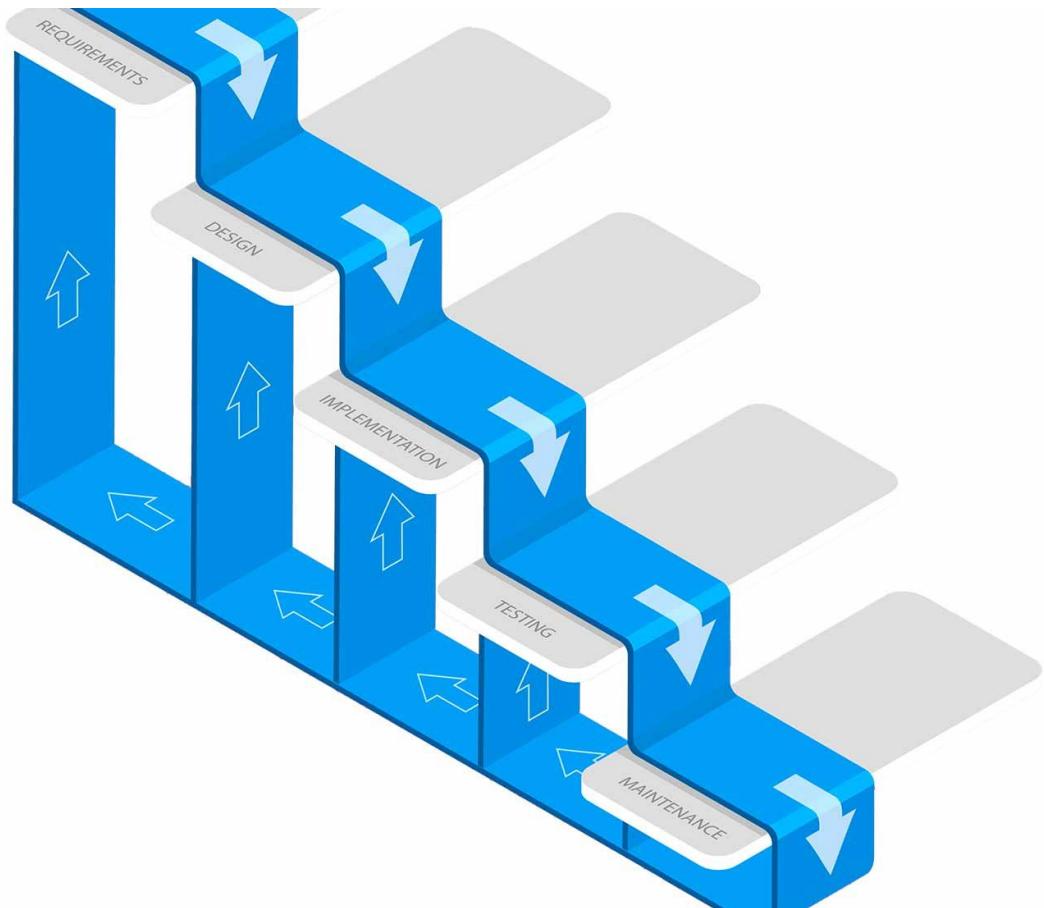
Waterfall Model



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

50

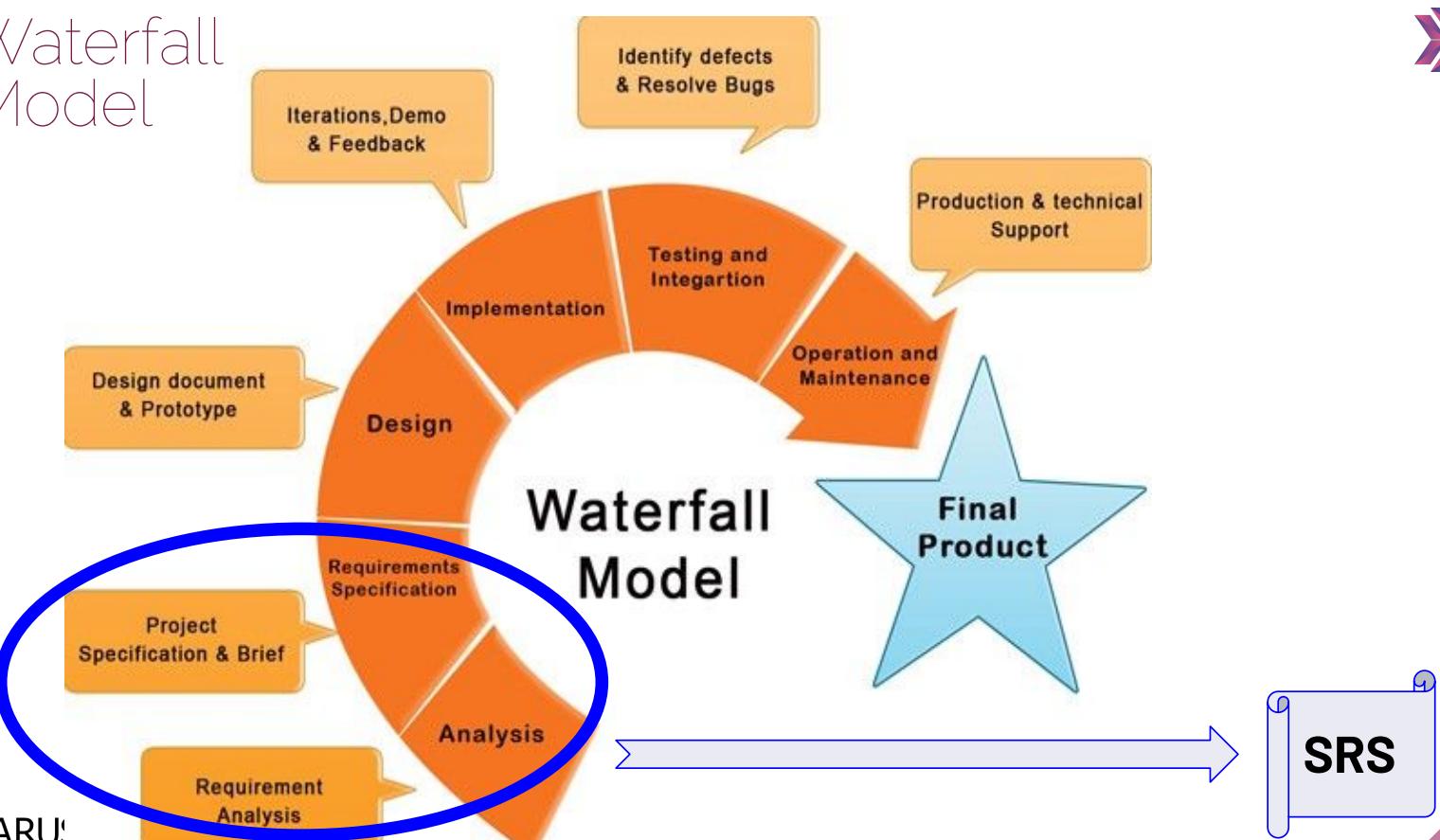
Waterfall ↗



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

51

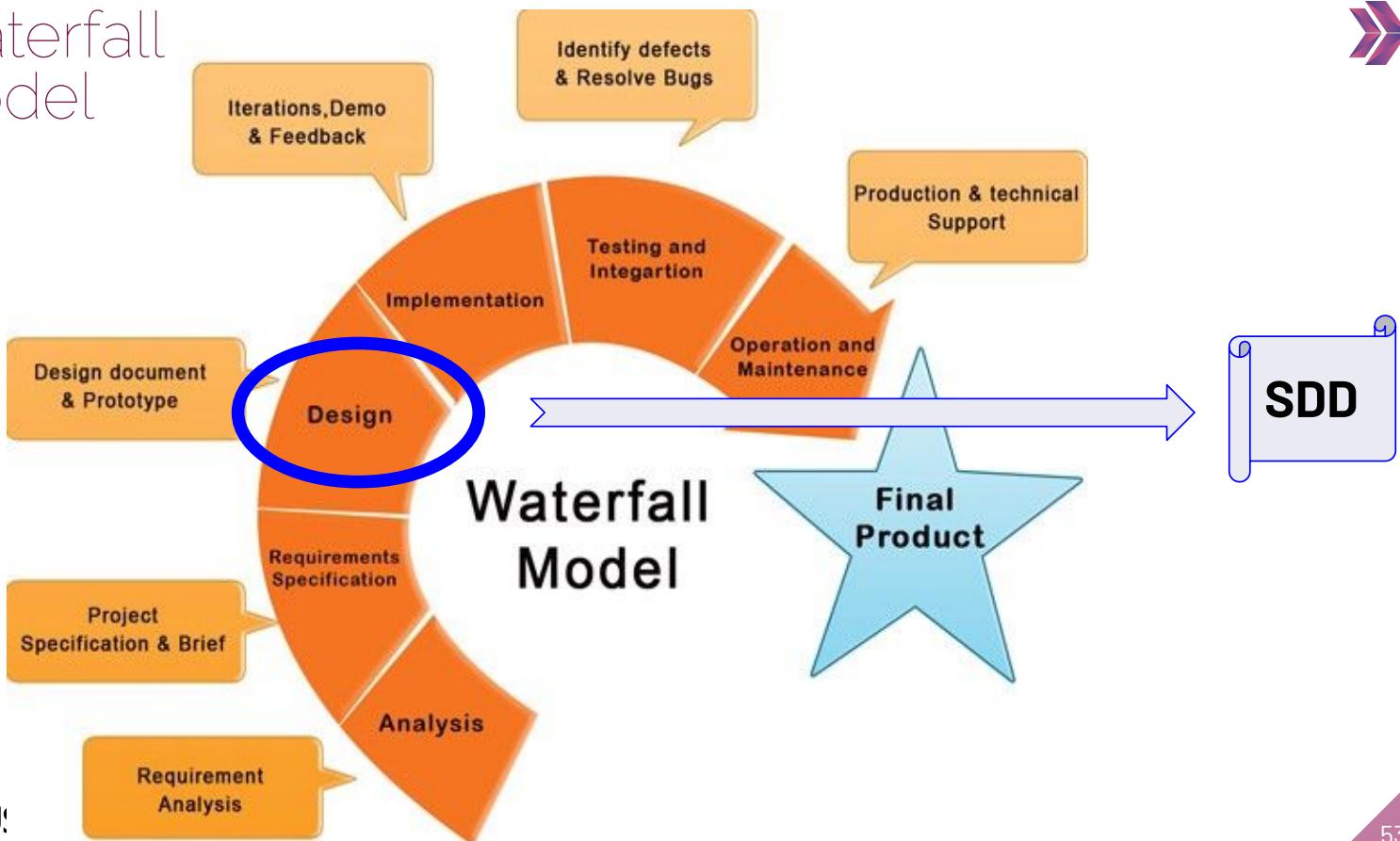
Waterfall Model



CLARU:
WAY TO REINVENT YOURSELF

52

Waterfall Model



CLARU!
WAY TO REINVENT YOURSELF

53

Waterfall Model

Coding is the other name for implementation/developing in SDLC.



Waterfall Model



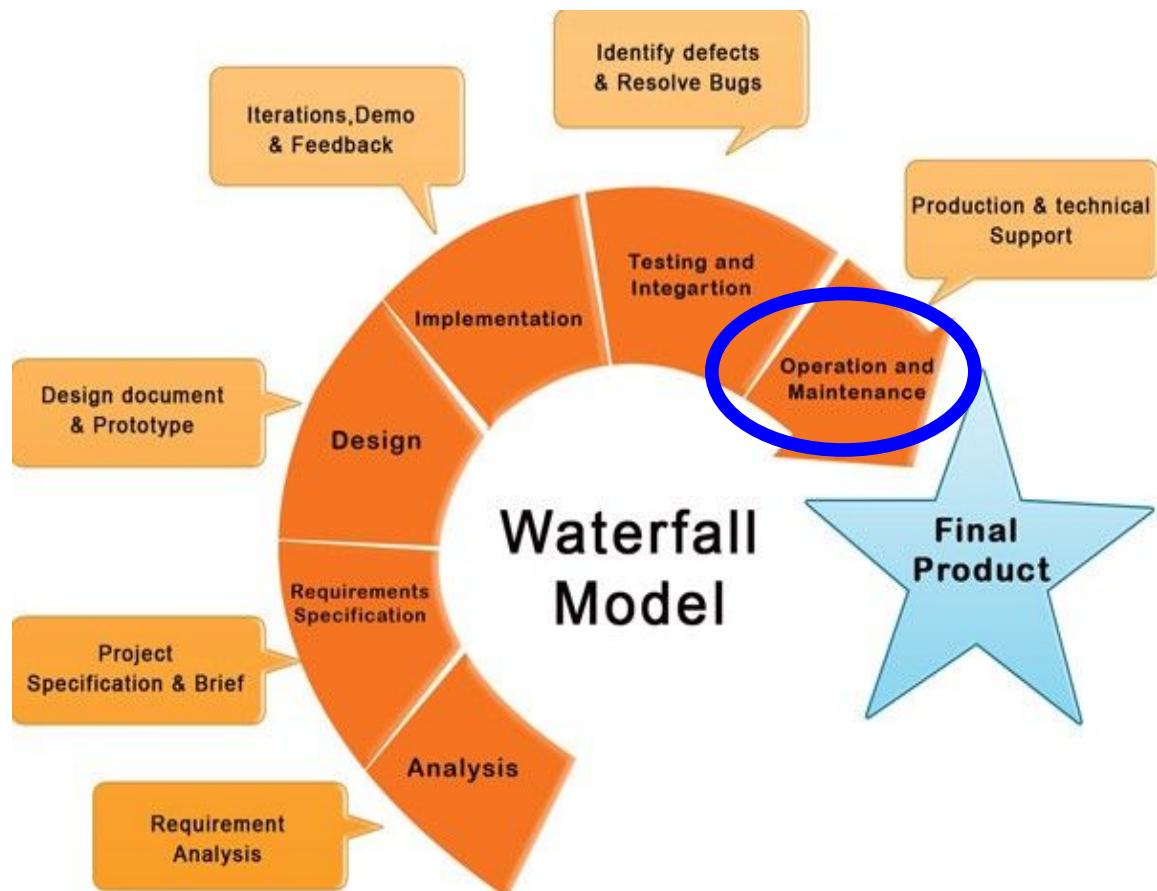
Waterfall Model



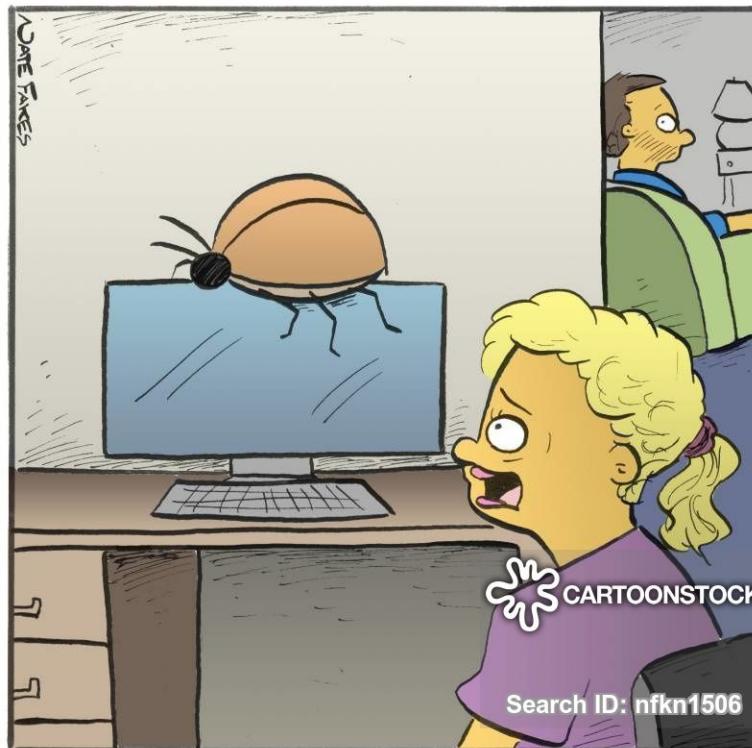
Waterfall Model



Waterfall Model



Waterfall Model



Hayatım, çabuk
geeelll!

Sanırım büyük bir
bug'ımız var.

*"Hon, come quick! I
think we have a major computer bug."*

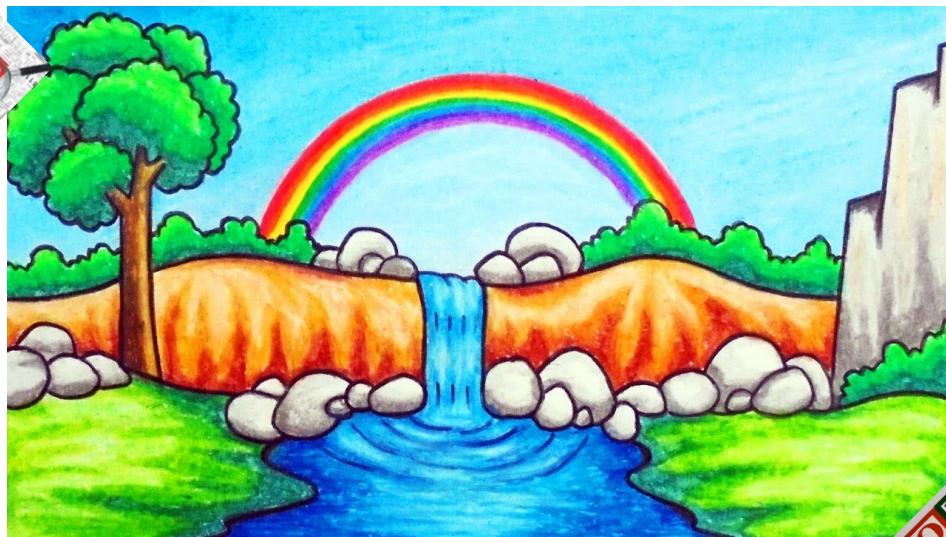
Application of the Waterfall Model





Waterfall Model

Advantages



Waterfall Model

disadvantages



Waterfall Model

disadvantages



63

Summary





THANKS!
Any questions?

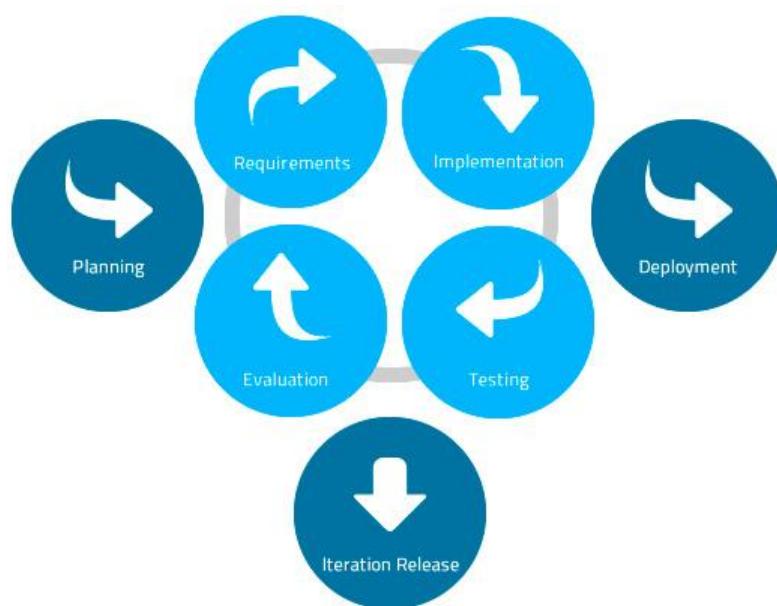


4

Iterative Model

CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Iterative Model

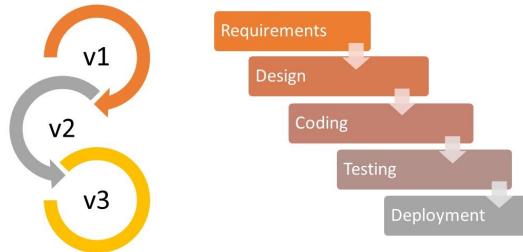


CLARUSWAY[©]
WAY TO REINVENT YOURSELF



Iterative Model

Iteration vs Waterfall



When we work **iteratively** we create rough product or product piece in one iteration

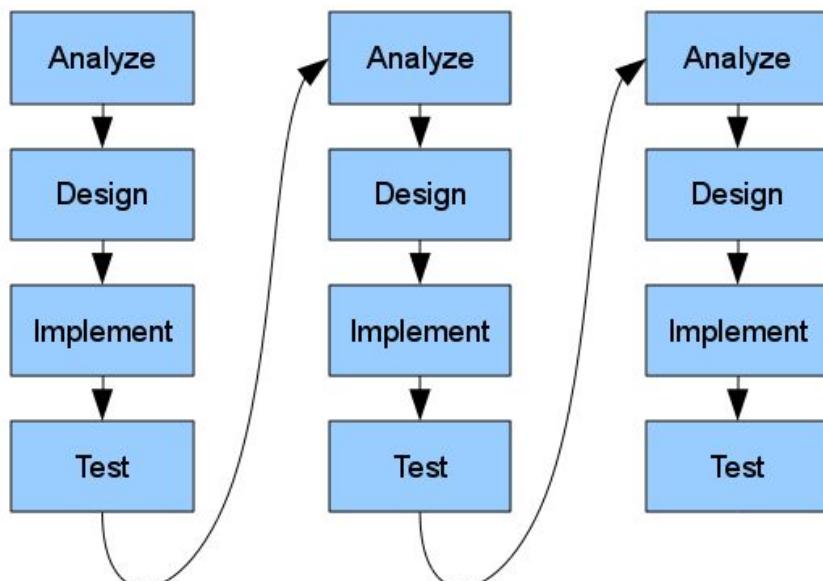
then review it and improve it in next iteration and so on until it's finished

- In the first iteration the whole painting is sketched roughly
- Then in the second iteration colors are filled
- In the third iteration finishing is done

The whole product is developed step by step

Iterative Model

Iteration 1



Iteration 2

Iteration 3

...Iteration N

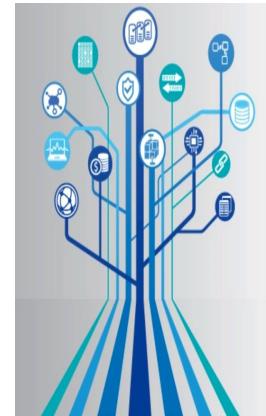


Model 1: Typical iterative development process



Iterative Model

advantages



Iterative Model

disadvantages





Iterative Model

Pros	Cons
<ul style="list-style-type: none"> ▪ Some working functionality can be developed quickly and early in the life cycle. ▪ Results are obtained early and periodically. ▪ Parallel development can be planned. ▪ Progress can be measured. ▪ Less costly to change the scope/requirements. ▪ Testing and debugging during smaller iteration is easy. ▪ Risks are identified and resolved during iteration; and each iteration is an easily managed milestone. ▪ Easier to manage risk - High risk part is done first. ▪ With every increment operational product is delivered. ▪ Issues, challenges & risks identified from each increment can be utilized/applied to the next increment. ▪ Risk analysis is better. ▪ It supports changing requirements. ▪ Initial Operating time is less. ▪ Better suited for large and mission-critical projects. ▪ During life cycle software is produced early which facilitates customer evaluation and feedback. 	<ul style="list-style-type: none"> ▪ More resources may be required. ▪ Although cost of change is lesser but it is not very suitable for changing requirements. ▪ More management attention is required. ▪ System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle. ▪ Defining increments may require definition of the complete system. ▪ Not suitable for smaller projects. ▪ Management complexity is more. ▪ End of project may not be known which is a risk. ▪ Highly skilled resources are required for risk analysis. ▪ Project's progress is highly dependent upon the risk analysis phase.

CLARUSWAY®
WAY TO REINVENT YOURSELF



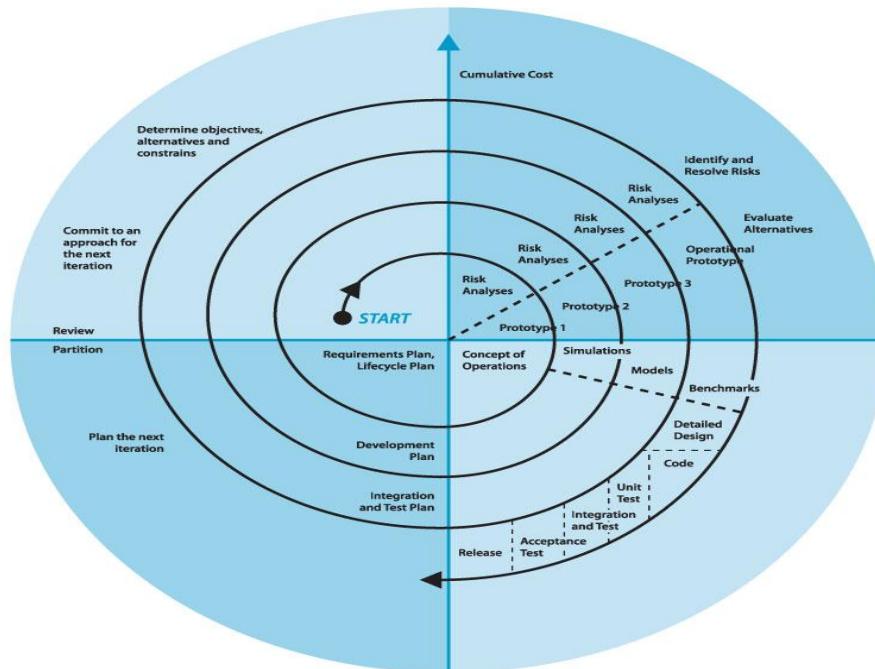
4

Spiral Model

CLARUSWAY®
WAY TO REINVENT YOURSELF



Spiral Model



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

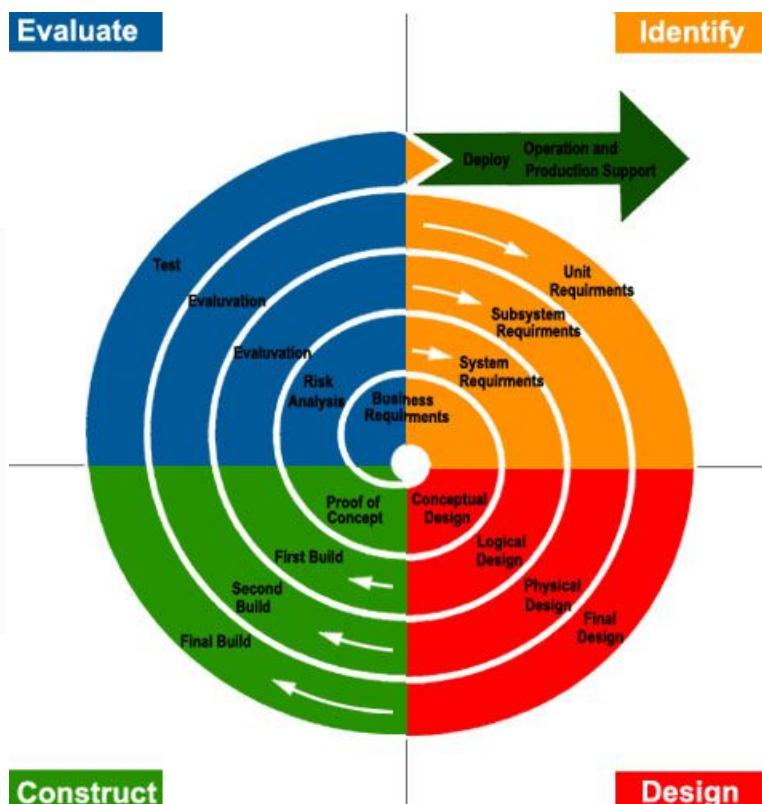
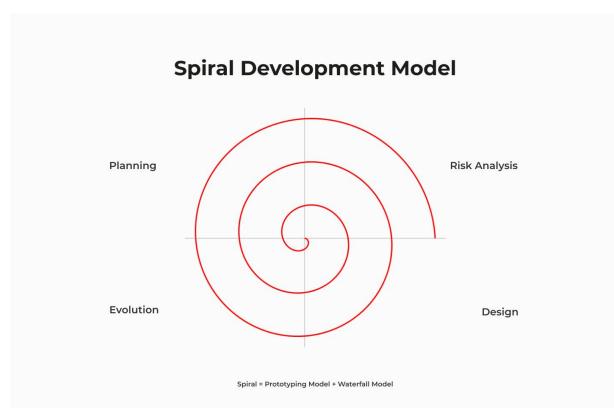
75



Spiral Model

Evaluate

Identify



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

76



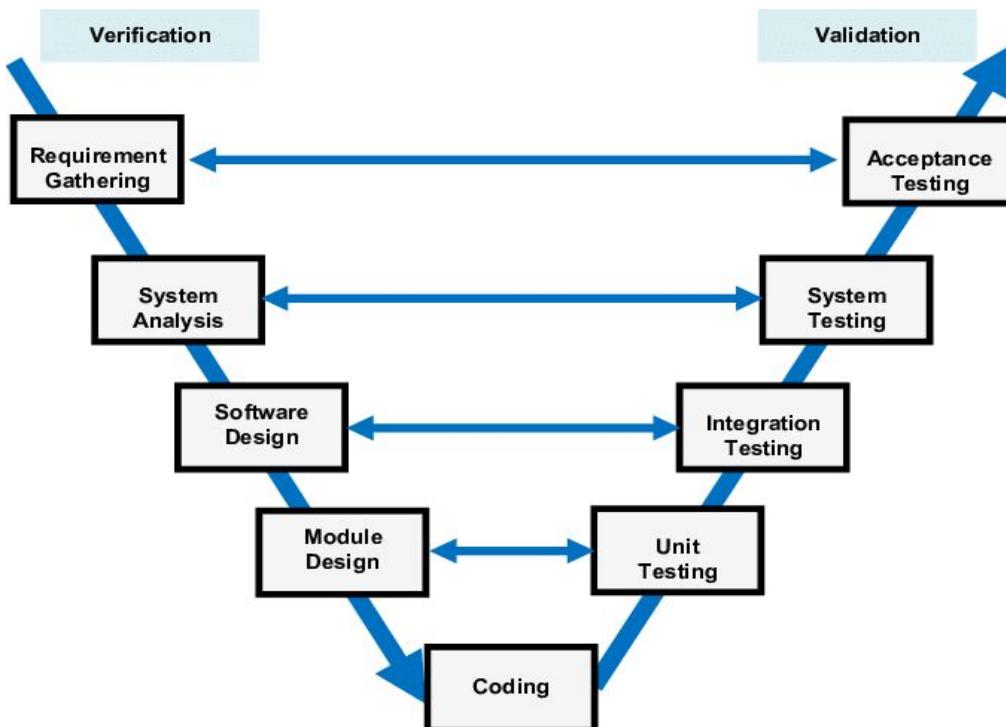
Spiral Model

Pros	Cons
<ul style="list-style-type: none"> ▪ Changing requirements can be accommodated. ▪ Allows for extensive use of prototypes ▪ Requirements can be captured more accurately. ▪ Users see the system early. ▪ Development can be divided into smaller parts and more risky parts can be developed earlier which helps better risk management. 	<ul style="list-style-type: none"> ▪ Management is more complex. ▪ End of project may not be known early. ▪ Not suitable for small or low risk projects and could be expensive for small projects. ▪ Process is complex ▪ Spiral may go indefinitely. ▪ Large number of intermediate stages requires excessive documentation.

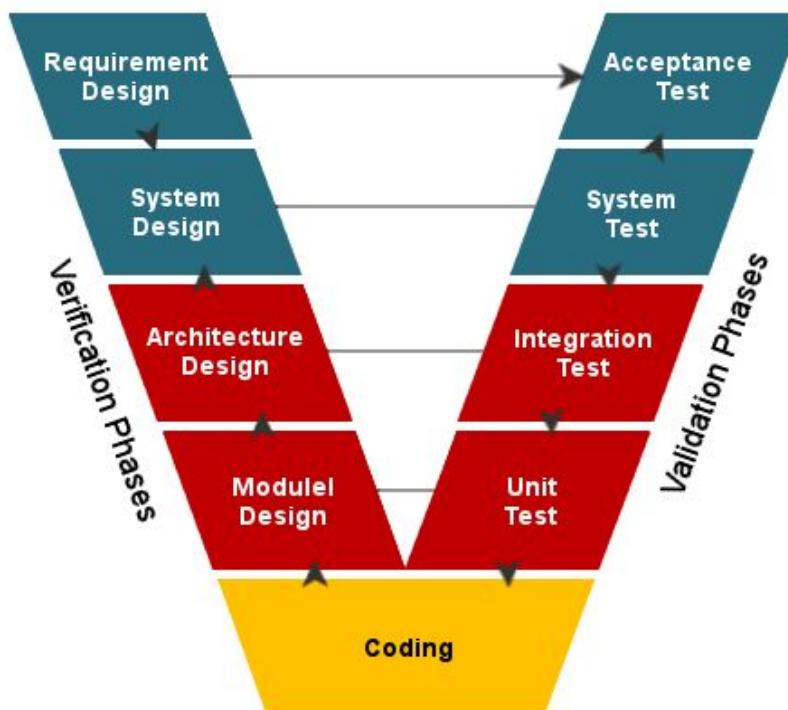


V - Model

V - Model



V - Model



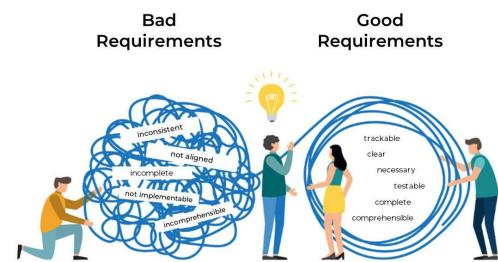
V - Model



advantages



Bad Requirements



Good Requirements

V - Model

disadvantages



V - Model



Pros	Cons
<ul style="list-style-type: none">• This is a highly disciplined model and Phases are completed one at a time.• Works well for smaller projects where requirements are very well understood.• Simple and easy to understand and use.• Easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.	<ul style="list-style-type: none">▪ High risk and uncertainty.▪ Not a good model for complex and object-oriented projects.▪ Poor model for long and ongoing projects.▪ Not suitable for the projects where requirements are at a moderate to high risk of changing.
	<ul style="list-style-type: none">▪ Once an application is in the testing stage, it is difficult to go back and change a functionality▪ No working software is produced until late during the life cycle.

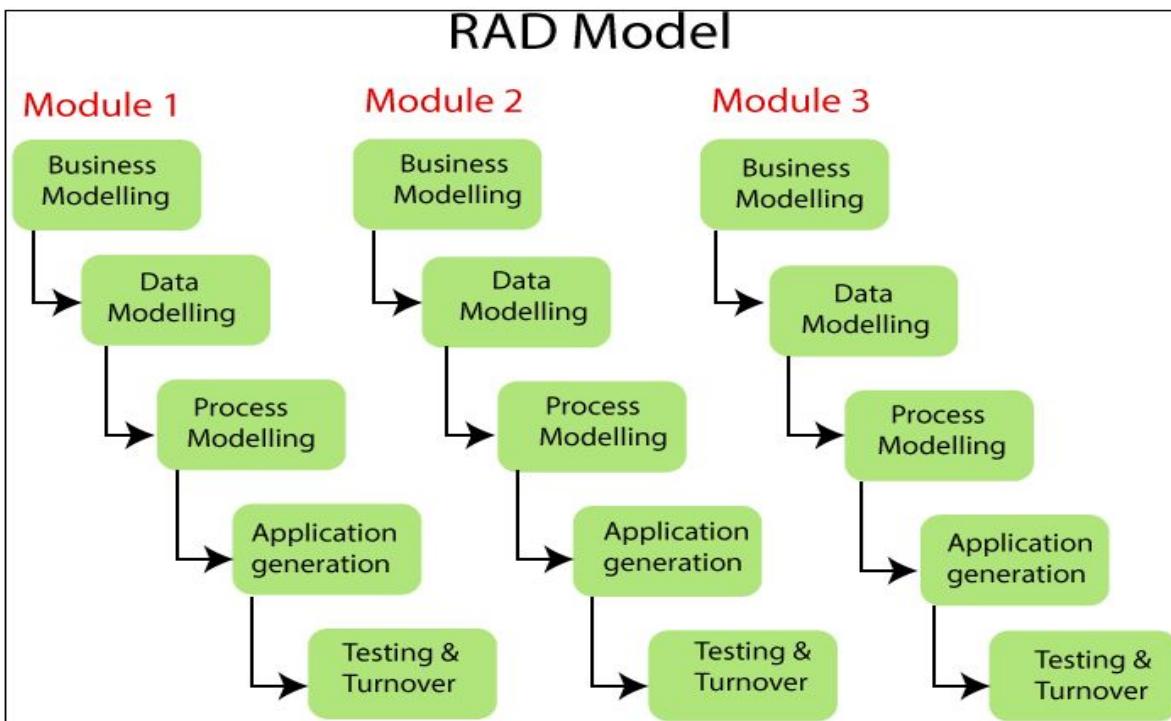
CLARUSWAY[©]
WAY TO REINVENT YOURSELF



4 RAD Model

CLARUSWAY[©]
WAY TO REINVENT YOURSELF

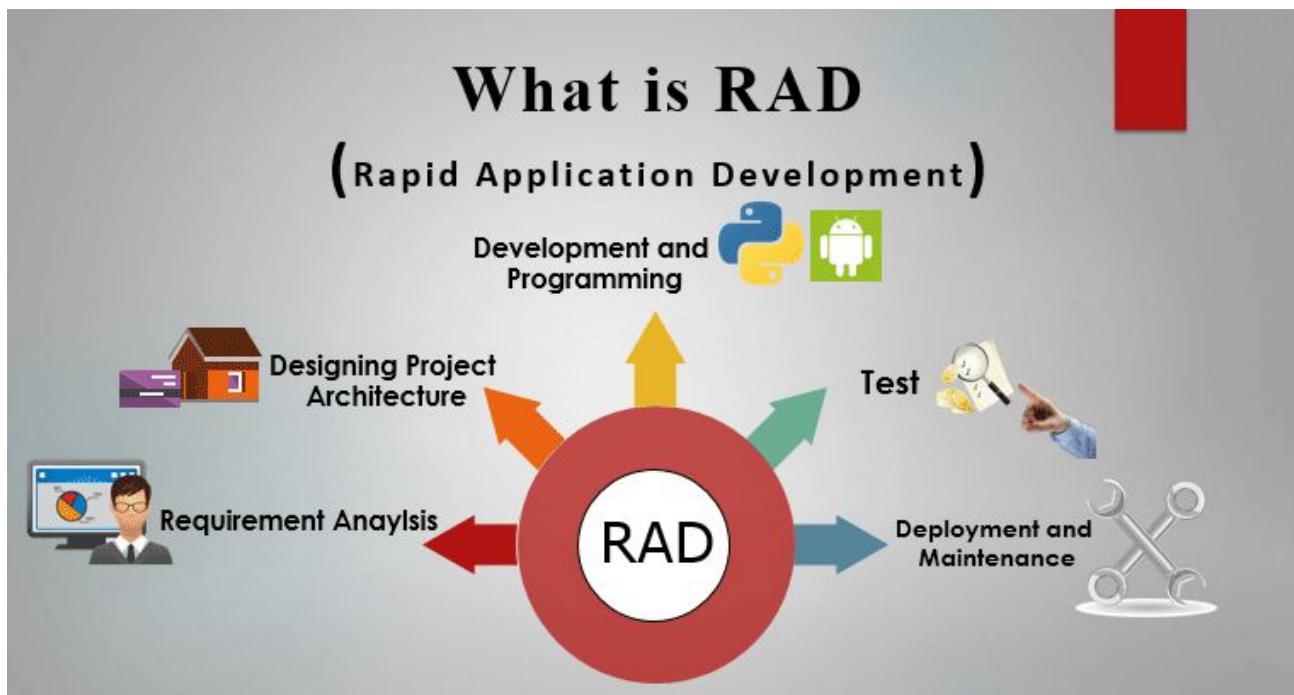
RAD Model



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

85

RAD Model



CLARUSWAY[©]
WAY TO REINVENT YOURSELF



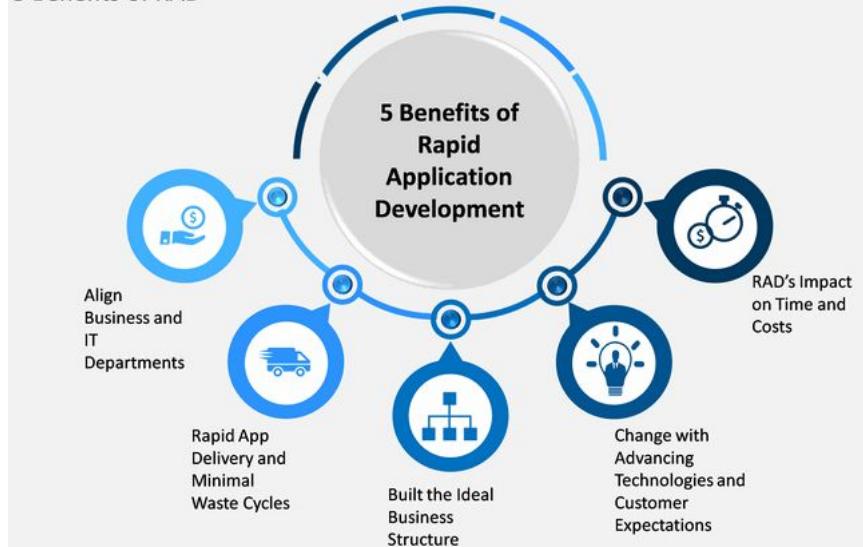
RAD Model

RAD



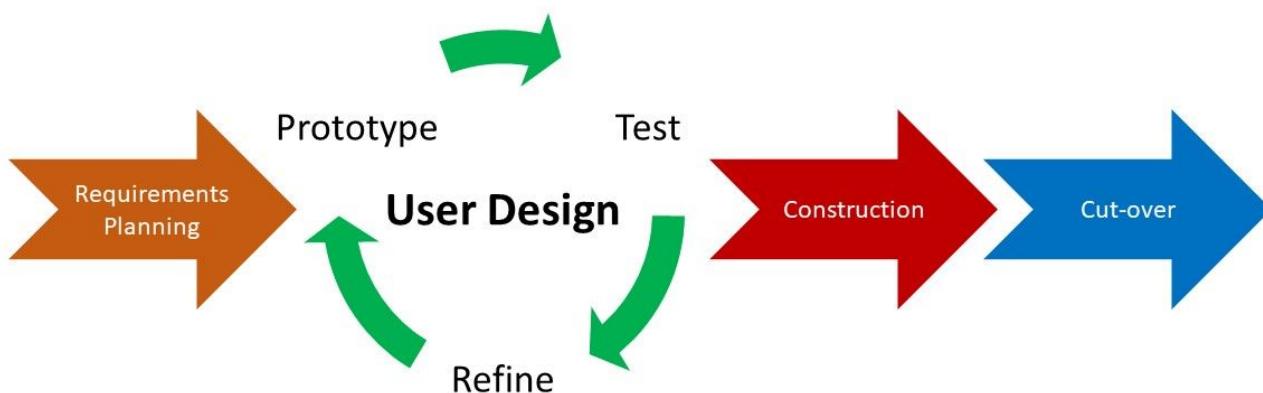
RAPID APPLICATION DEVELOPMENT

5 Benefits of RAD



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

RAD Model



CLARUSWAY[©]
WAY TO REINVENT YOURSELF



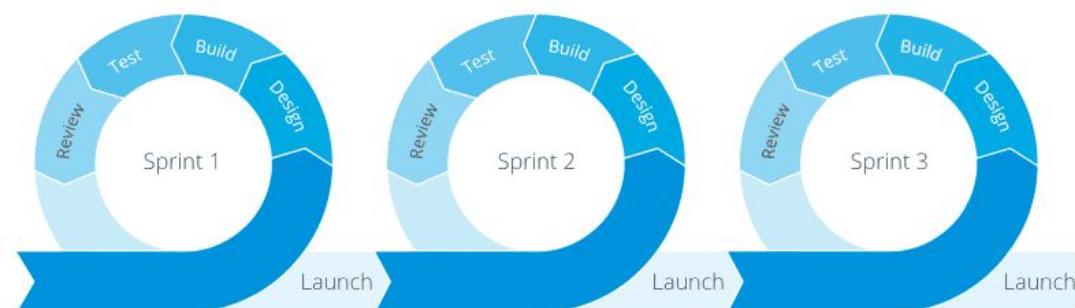
RAD Model

Pros	Cons
<ul style="list-style-type: none"> ▪ Changing requirements can be accommodated. ▪ Progress can be measured. ▪ Iteration time can be short with use of powerful RAD tools. ▪ Productivity with fewer people in short time. ▪ Reduced development time. ▪ Increases reusability of components 	<ul style="list-style-type: none"> ▪ Dependency on technically strong team members for identifying business requirements. ▪ Only system that can be modularized can be built using RAD ▪ Requires highly skilled developers/designers. ▪ High dependency on modeling skills ▪ Inapplicable to cheaper projects as cost
<ul style="list-style-type: none"> ▪ Quick initial reviews occur ▪ Encourages customer feedback ▪ Integration from very beginning solves a lot of integration issues. 	<ul style="list-style-type: none"> of modeling and automated code generation is very high. ▪ Management complexity is more. ▪ Suitable for systems that are component based and scalable. ▪ Requires user involvement throughout the life cycle. ▪ Suitable for project requiring shorter development times.



Agile Model

Agile Model



Each iteration lasts from 1 - 3 weeks



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

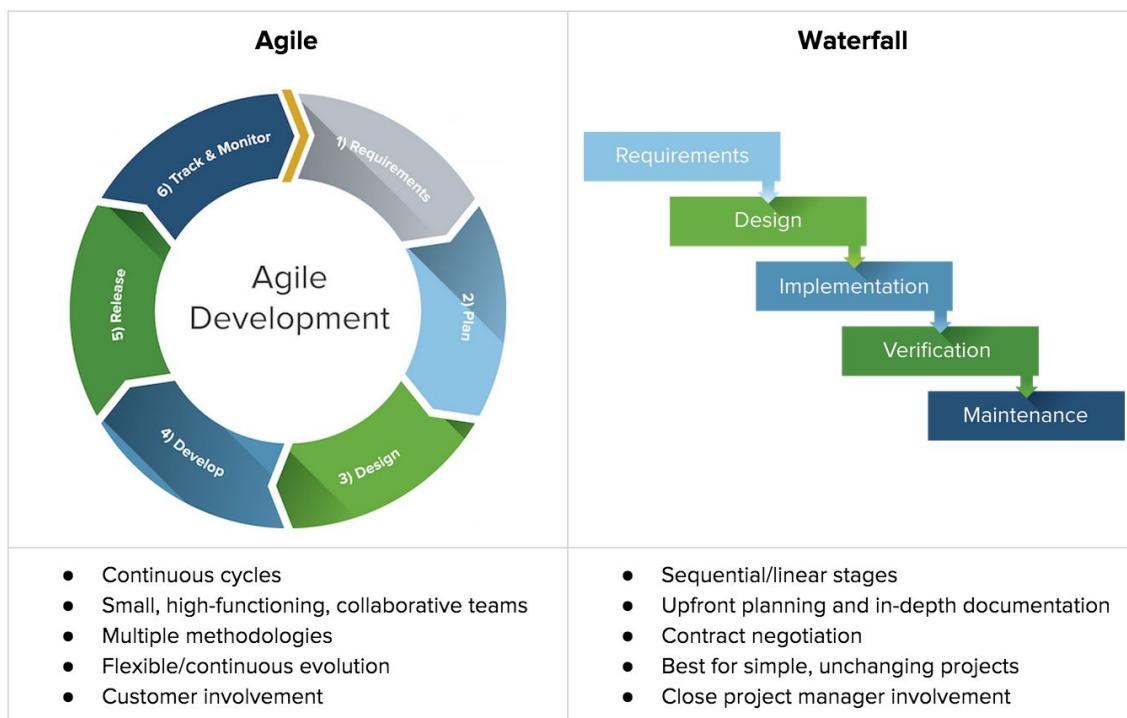
Agile Model



AGILE SOFTWARE DEVELOPMENT

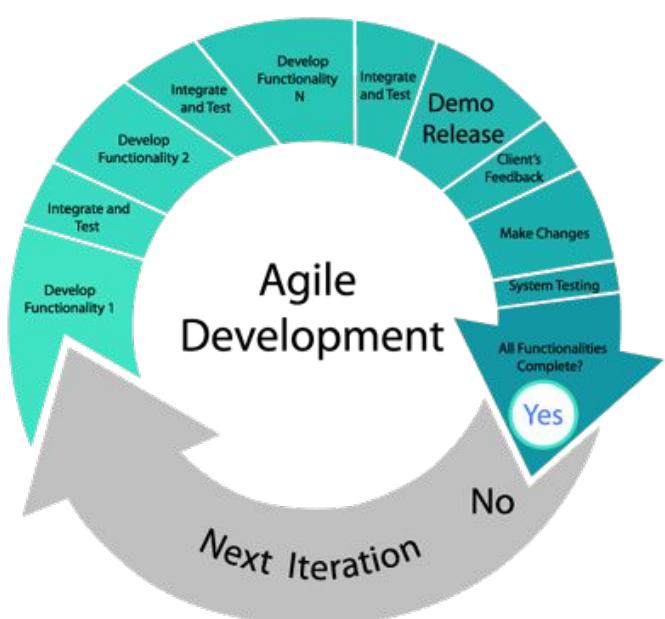
CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Agile Vs Traditional SDLC Models



CLARUSWAY[©]
WAY TO REINVENT YOURSELF

Agile Vs Traditional SDLC Models



CLARUSWAY[©]
WAY TO REINVENT YOURSELF



THANKS!

Any questions?