Mansoura University
Faculty of Computers and Information
Department of Information System
First Semester- 2025-2026

[Agile Methods

Prof. Samir Abdelrazek

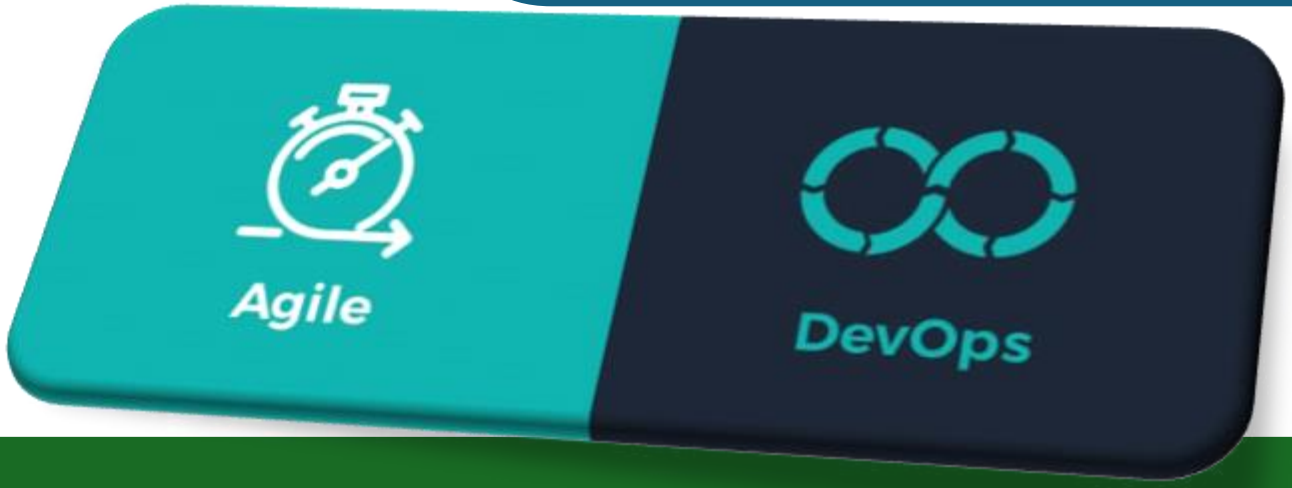
Lecture 2 System Development Methods

Prof. Samir Abd ElRazek

2025

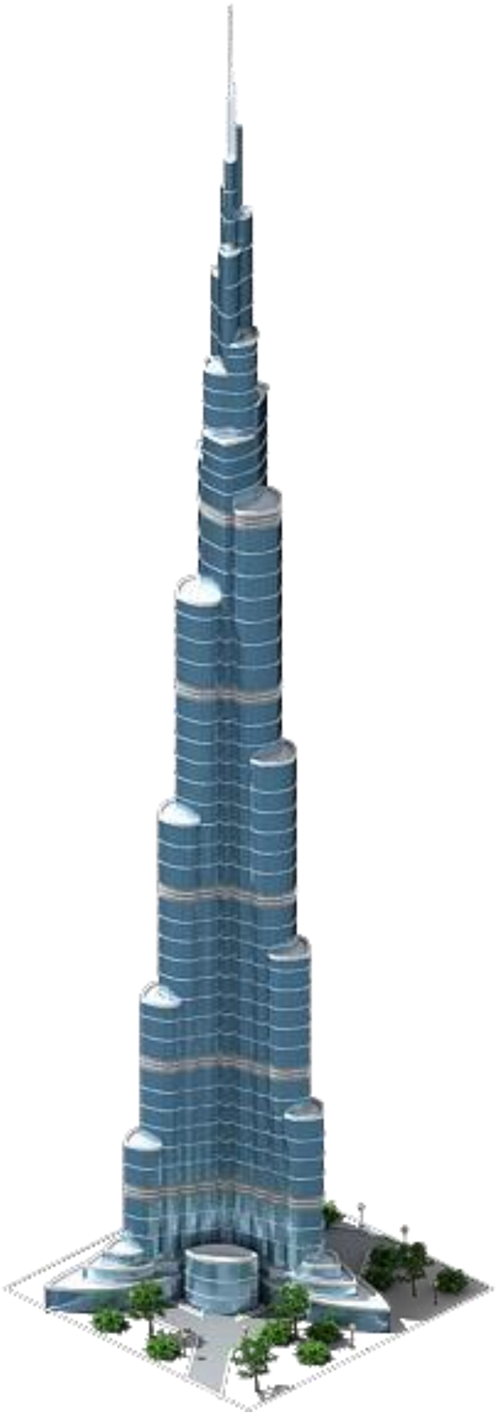
Contents

- Structured Analysis
- Object-Oriented Analysis
- Phased and Prototyping Methods (JAD, RAD)
- Agile Development and DevOps techniques
- Criteria of Selecting a Method



Old Approach
Structured Analysis

Structured analysis is a traditional systems development technique that is time-tested and easy to understand.



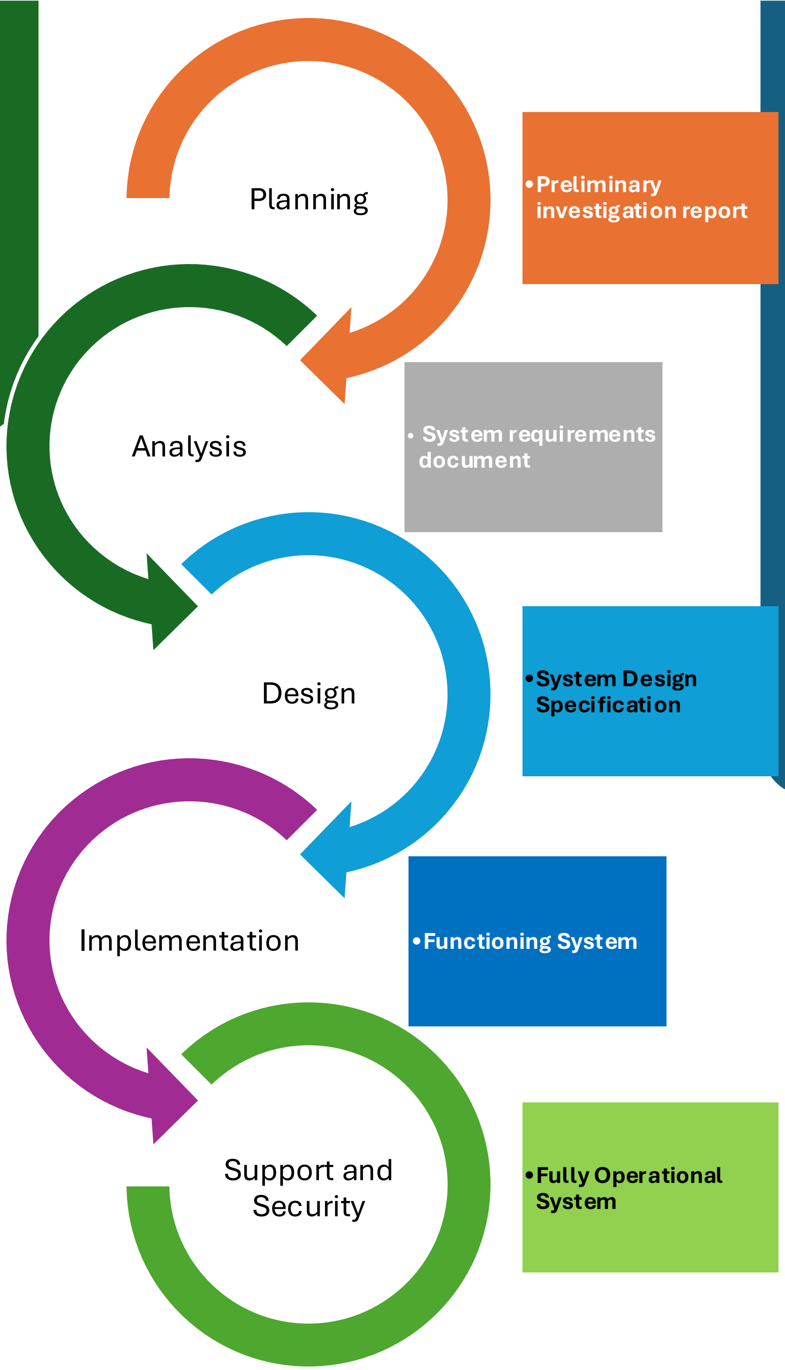
Structured Analysis

Structured analysis uses a series of phases, called the systems development life cycle (SDLC), to plan, analyze, design, implement, and support an information system.

based on an overall plan, similar to a blueprint for constructing a building, so it is called a predictive approach.

Use 2 Approaches:

- **Waterfall model**
- **Parallel model**


Planning
•Preliminary investigation report
Analysis
• System requirements document
Design
•System Design Specification
Implementation
•Functioning System
Support and Security
•Fully Operational System

Waterfall Model

In the waterfall model, the result of each phase is called a deliverable, which flows into the next phase.

Waterfall and parallel models does not emphasize interactivity among the phases.

The SDLC model usually includes five steps

System Planning

- Begins with a formal request to the IT department, called a systems request.
- The purpose of this phase is to perform a preliminary investigation to evaluate an IT-related business opportunity or problem.
- A key part of the preliminary investigation is a feasibility study that reviews anticipated costs and benefits

System Analysis

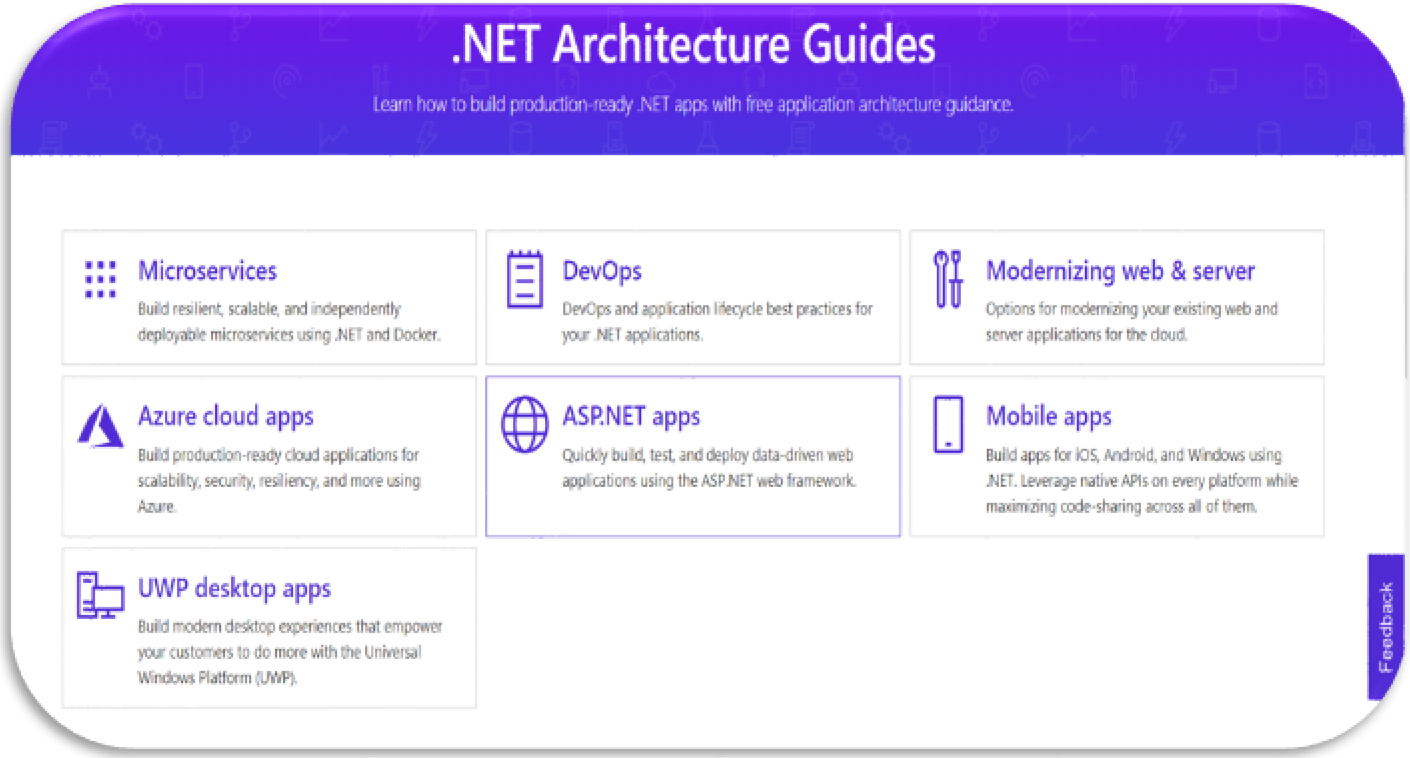
Required for building a logical model of the new system.

- The first step is requirements modeling, where the analyst investigates business processes.
- The deliverable for the systems analysis phase is the system requirements document.

System Design

Create a physical model that will satisfy all documented requirements for the system.

- The user interface is designed and necessary outputs, inputs, and processes are identified.
- The application architecture is also determined, which programmers will use to transform the logical design into program modules and code.



.NET Architecture Guides

Learn how to build production-ready .NET apps with free application architecture guidance.

Microservices
Build resilient, scalable, and independently deployable microservices using .NET and Docker.

DevOps
DevOps and application lifecycle best practices for your .NET applications.

Modernizing web & server
Options for modernizing your existing web and server applications for the cloud.

Azure cloud apps
Build production-ready cloud applications for scalability, security, resiliency, and more using Azure.

ASP.NET apps
Quickly build, test, and deploy data-driven web applications using the ASP.NET web framework.

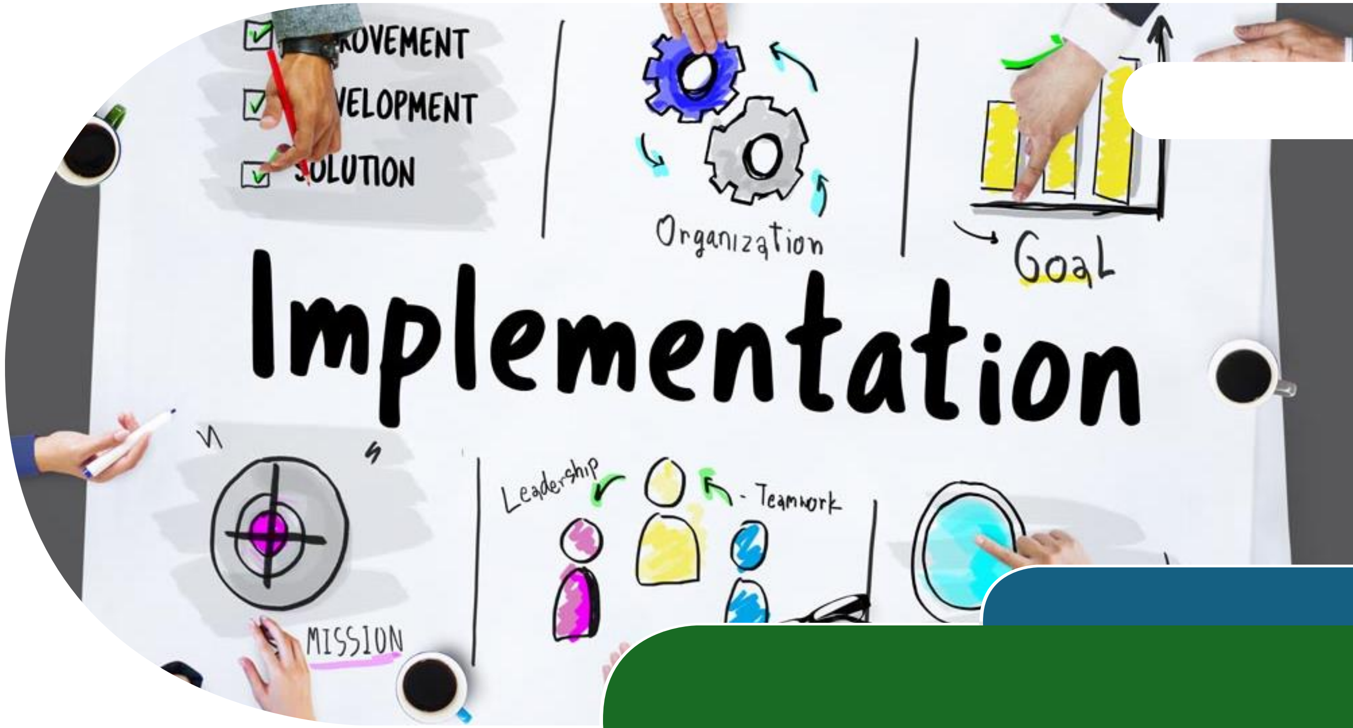
Mobile apps
Build apps for iOS, Android, and Windows using .NET. Leverage native APIs on every platform while maximizing code-sharing across all of them.

UWP desktop apps
Build modern desktop experiences that empower your customers to do more with the Universal Windows Platform (UWP).

System Implementation

- The objective of the systems implementation phase is to deliver a completely **functioning and documented information system.**

- **The system is ready for use. Final preparations include converting data to the new system's files, etc…**
- **systems evaluation: determine whether the system operates properly and if costs and benefits are within expectations.**



System Support & Security

- IT staff maintains, enhances, and protects the system. **Maintenance** changes correct errors and adapt to changes in the environment

- **The objective during this phase is to maximize return on the IT investment (ROI).**
- **A well-designed system must be secure, reliable, maintainable, and scalable.**

Waterfall Model

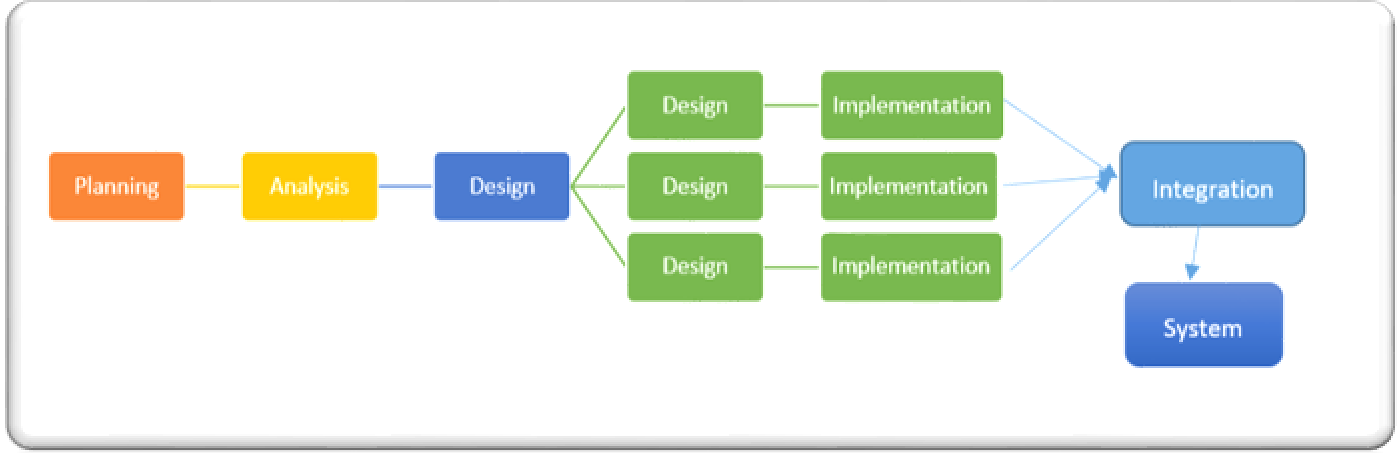
Pros

- Definitely not a great choice complex and long-term project
- Impossible to make changes at the later stage
- This SDLC model doesn't work for maintenance type project
- Successful execution of the project depends on precise requirements
- No working model of the software until the end of the life cycle

Cons

- **Waterfall model is easy to understand due to its simple linear structure**
- **It helps to define the goals and deliverables at the early stage of the project**
- **Project management is simple and effective as there are no unnecessary changes**
- **This model works effectively with smaller projects where the requirements are clearly**
- **Each stage are clearly defined and understood**

Parallel Model

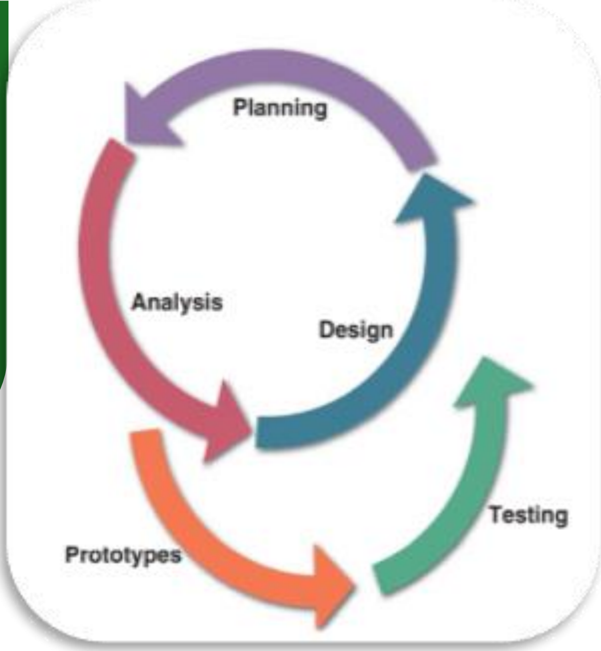


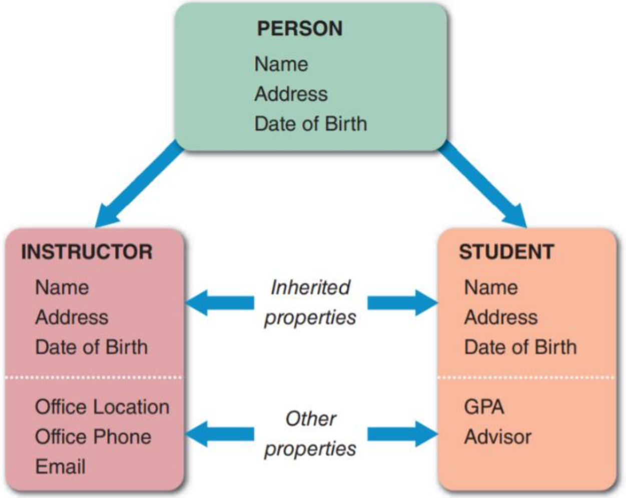
Cutting the Project down to subprojects, Parallel development methodology attempts to address the problem of long delays between the analysis phase and the delivery of the system.

Reduce the time to deliver a system; thus, there is less chance of changes in the business environment causing rework.

Object-Oriented Analysis

The result is a set of software objects that represent actual people, things, transactions, and events. Using an O-O programming language, a programmer then writes the code that creates the objects.





PERSON
Name
Address
Date of Birth

INSTRUCTOR
Name
Address
Date of Birth
Office Location
Office Phone
Email

Inherited properties

Other properties

STUDENT
Name
Address
Date of Birth
GPA
Advisor

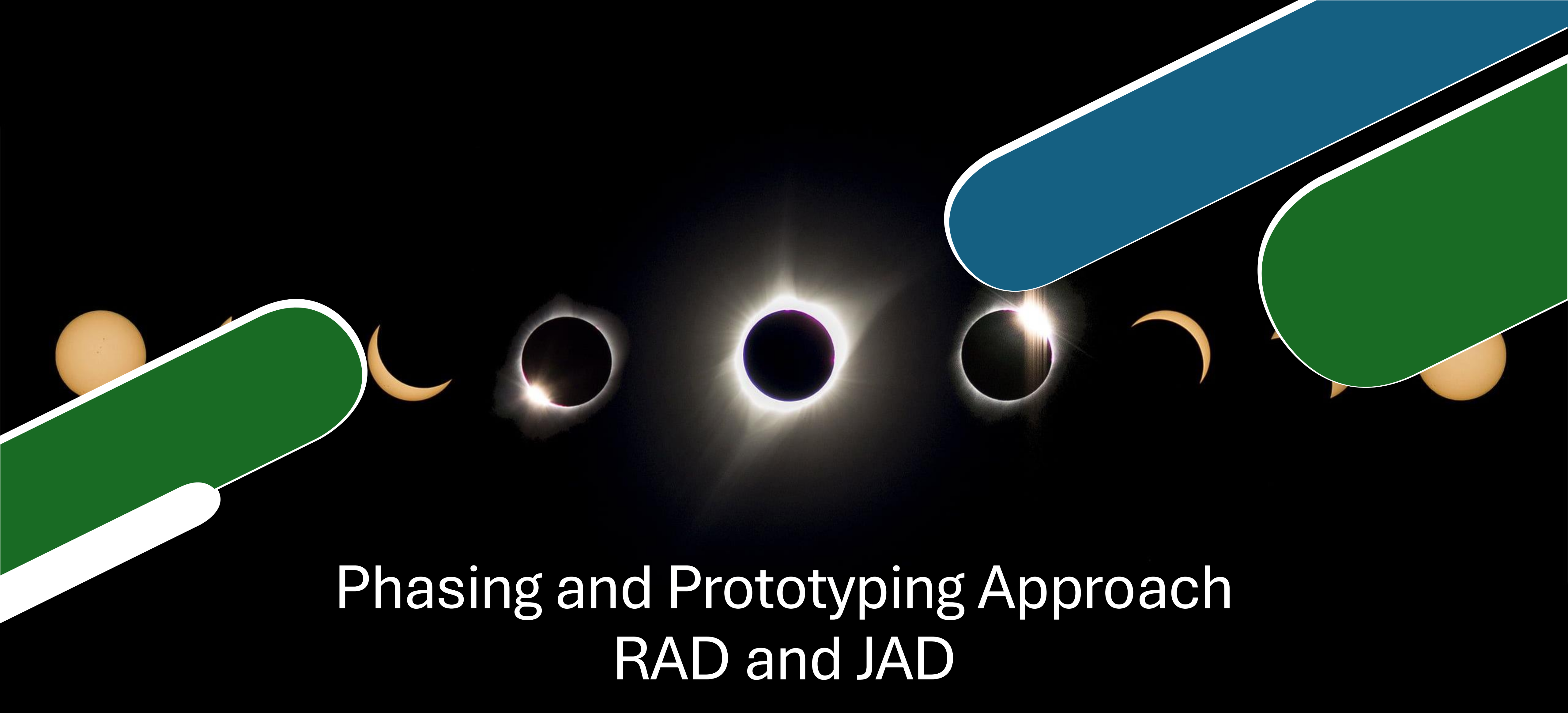
Object-Oriented Model

Structured analysis treats processes and data as separate components.

object-oriented analysis combines data and the processes that act on the data into things called objects

This concept about how to develop with some special design patterns, as the class is a blueprint of the objects

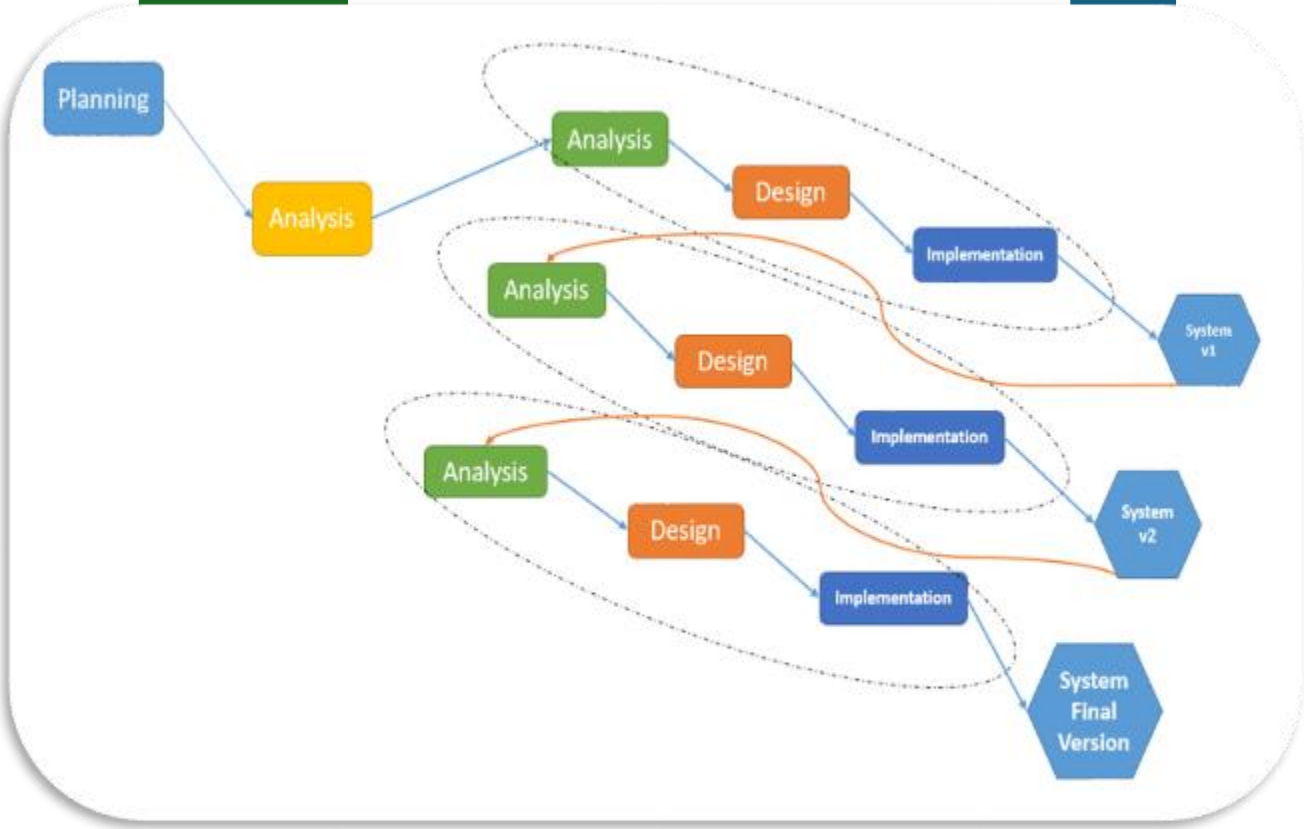
The classes are the deliverables as a prototypes as it phased, example of inheritance as a concept of O-O



Phasing and Prototyping Approach RAD and JAD

To provide a more accurate system, the companies manipulate the SDLC to create new versions of methodologies.

PHASED DEVELOPMENT

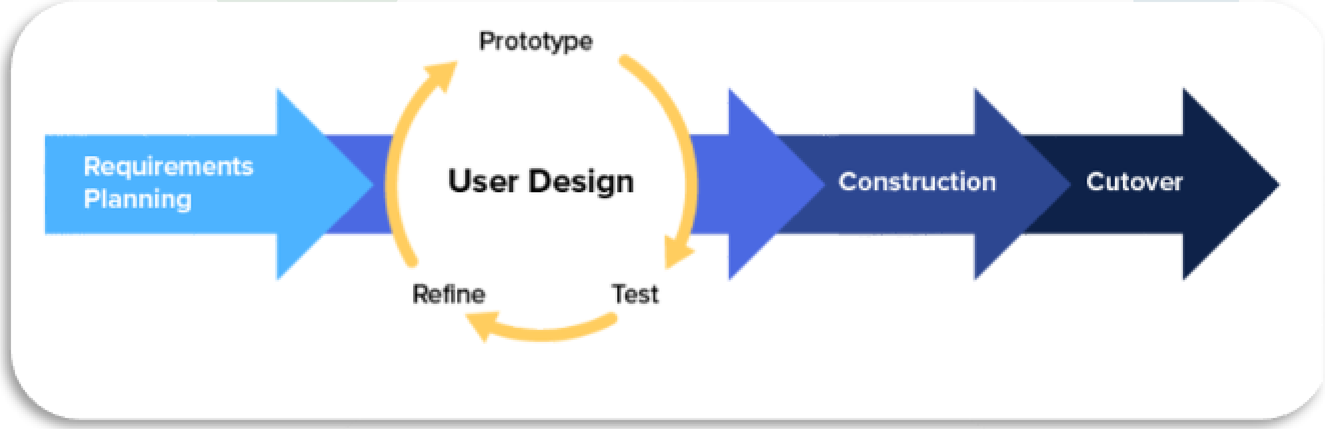


Once version 1 is implemented, work begins on version 2. Additional analysis is performed based on the previously then completing other phases

breaks an overall system into a series of versions that are developed sequentially.

The most important and fundamental requirements are bundled into the first version of the system; this is a type of RAD development explained in next slide.

RAPID APPLICATION DEVELOPMENT (RAD)



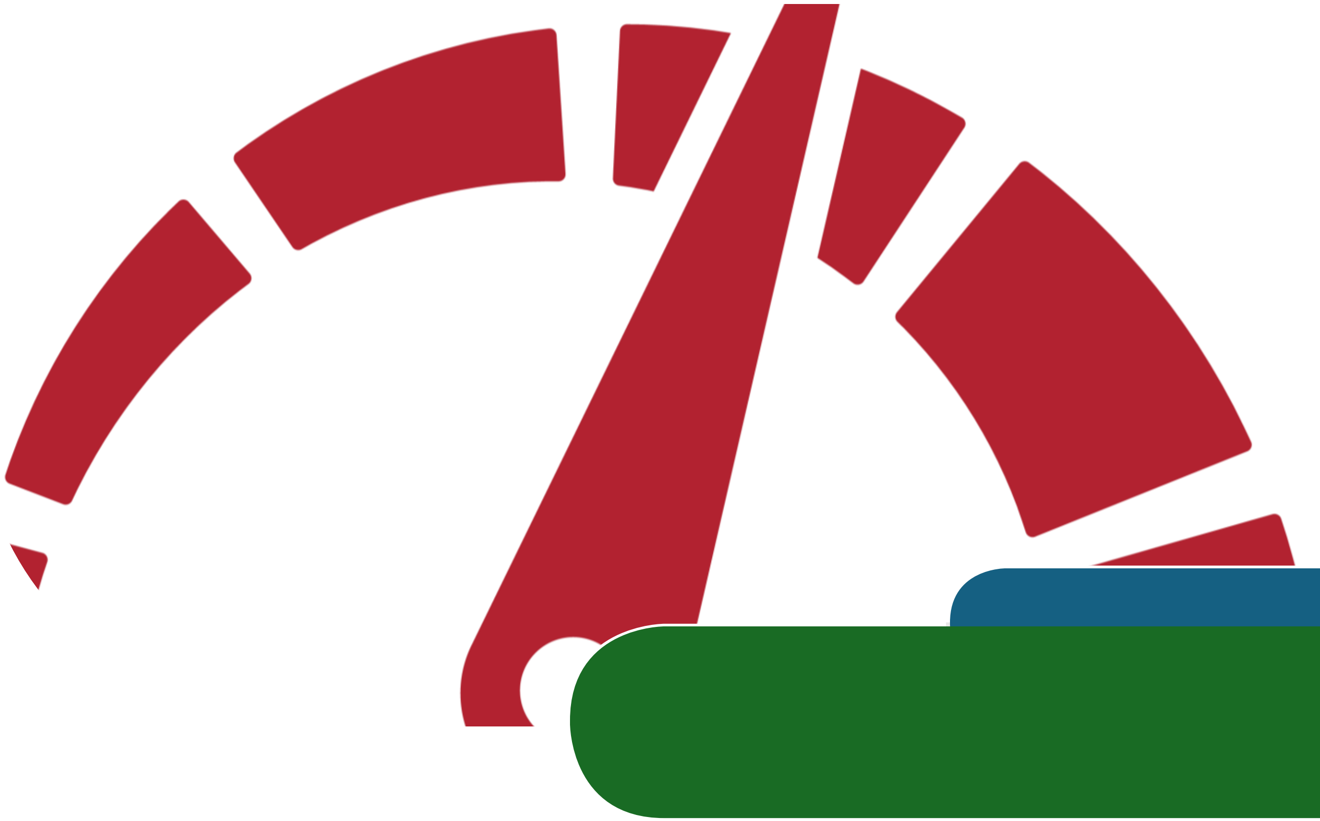
RAD-based methodologies attempt to address both weaknesses of structured design methodologies by adjusting the SDLC phases.

system developed quickly and into the hands of the users

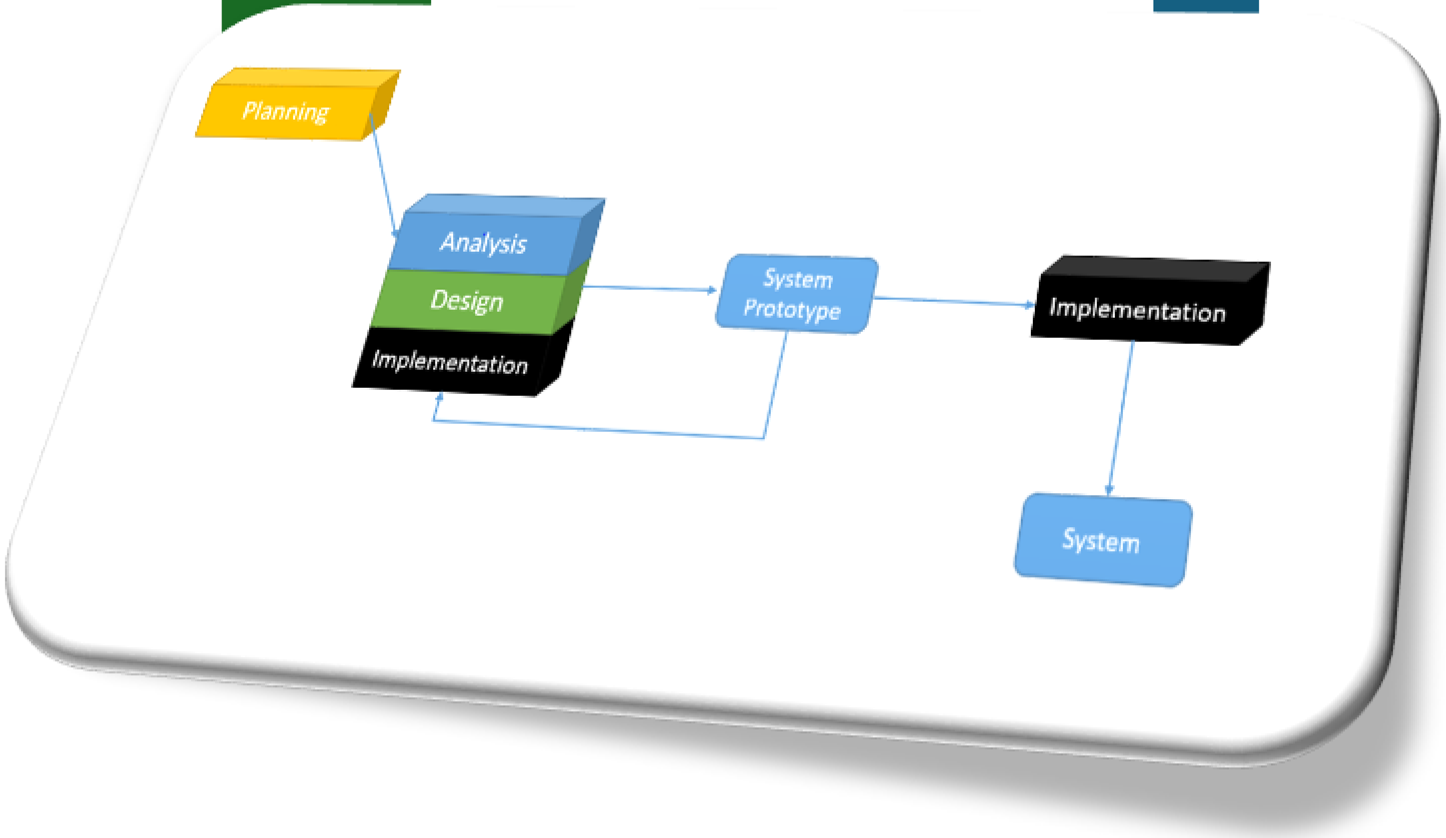
RAD-based methodologies recommend that analysts use special techniques such as computer-aided software engineering (CASE) tools, joint application design (JAD) sessions, fourth-generation or visual programming languages

RAD VS JAD

- The difference is that JAD focuses on team-based fact-finding, which is only one phase of the development process, whereas RAD is more like a compressed version of the entire process.



Prototyping DEVELOPMENT



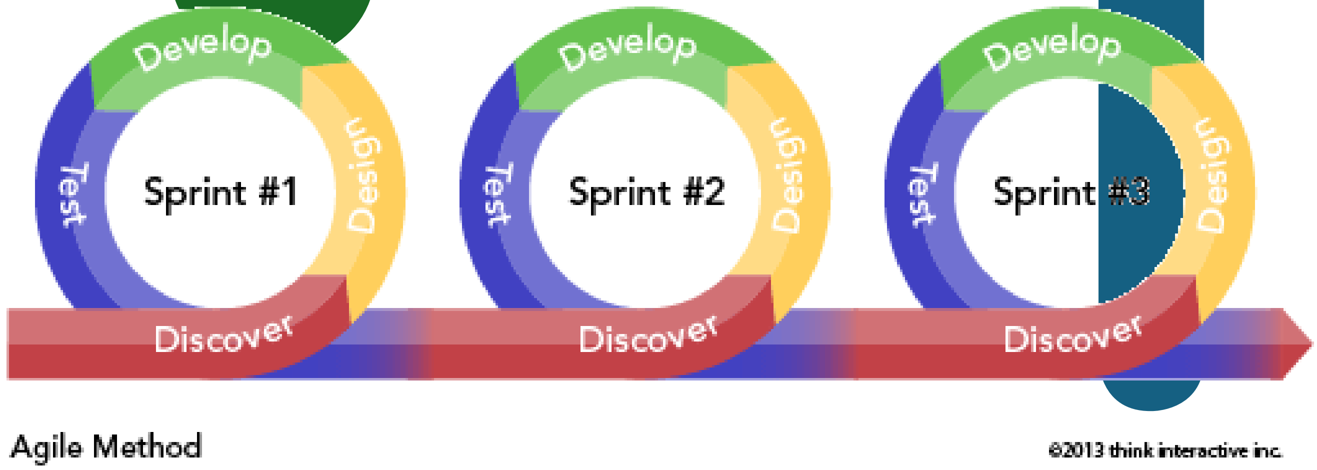
The key advantage of a prototyping-based methodology is that it very quickly provides a system with which the users can interact

Performs the analysis, design, and implementation phases concurrently, and all three phases are performed repeatedly in a cycle

the basics of analysis and design are performed, and work immediately begins on a system prototype. A quick-and-dirty program that provides a minimal amount of features.

Agile Development and DevOps Techniques

An agile approach emphasizes continuous feedback, and each incremental step is affected by what was learned in the prior steps.



Agile methods typically use a spiral model, which represents a series of iterations, or revisions, based on user feedback.

Agile Methods

Structured analysis builds an overall plan for the information system, just as a contractor might use a blueprint for constructing a building.

Agile methods, in contrast, attempt to develop a system incrementally, by building a series of prototypes and constantly adjusting them to user requirements

Agile Manifesto

Reflect Tune Adjust
WORK TOGETHER DAILY
Business People AND Developers
SUSTAINABLE DEVELOPMENT
CONSTANT PACE
Self Organising TEAMS
HARNESS change
HIGHEST PRIORITY
EARLY + CONTINUOUS DELIVERY
Changing REQUIREMENTS are WELCOME
AGILE manifesto
Satisfy
CUSTOMER
SIMPLICITY
DELIVER frequently
SHORTER Timescale
CONTINUOUS ATTENTION
TECHNICAL EXCELLENCE
BUILD PROJECTS around MOTIVATED individuals
environment
trust
support
FACE to FACE IS MOST EFFICIENT EFFECTIVE
conveying INFORMATION
IT WORKS
Measure of PROGRESS
GOOD DESIGN
Graphic Recording BY Lynne Cazaly
www.lynnecazaly.com
iPAD JUNE 9, 2012

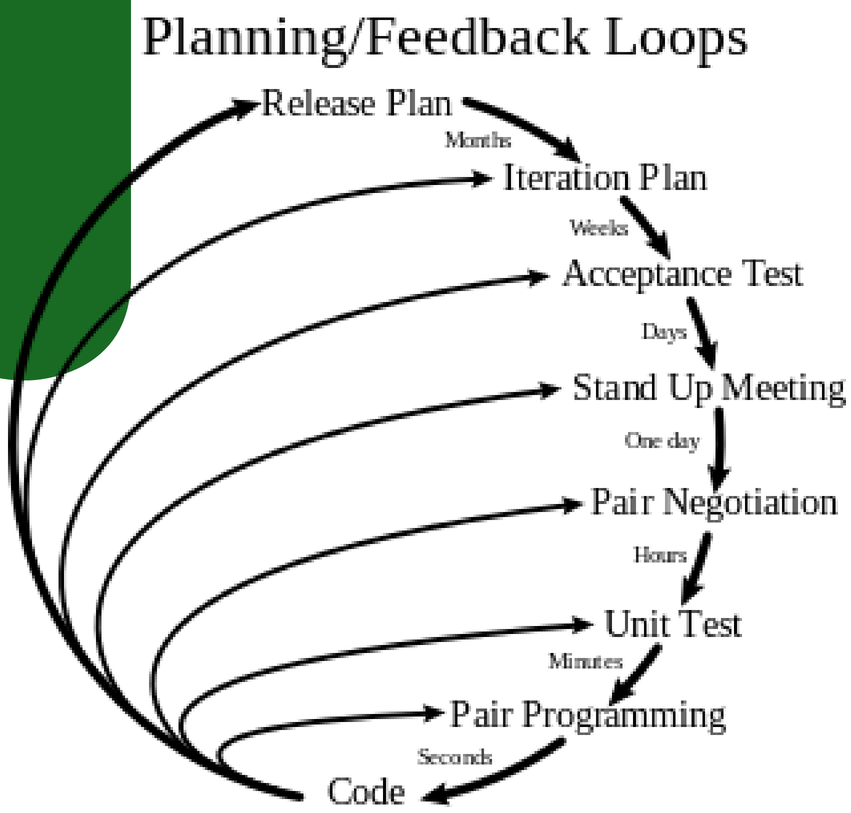
Nearly 20 years ago, 17 software developers came together in Snowbird, Utah to propose a new way of developing software "by doing it and helping others do it."

The Agile Manifesto is a document that identifies four key values and 12 principles that its authors believe software developers should use to guide their work.

Key Values:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

XP EXTREME PROGRAMMING



Founded on four core values: communication, simplicity, feedback, and courage.

An XP project begins with user stories that describe what the system needs to do.

Then, programmers code in small, simple modules and test to meet those needs.

XP projects deliver results sooner than even the RAD approaches, and they rarely get bogged down in gathering requirements for the system

SCRUM

Scrum is a term that is well known to rugby fans. In rugby, a scrum is used to restart a game.

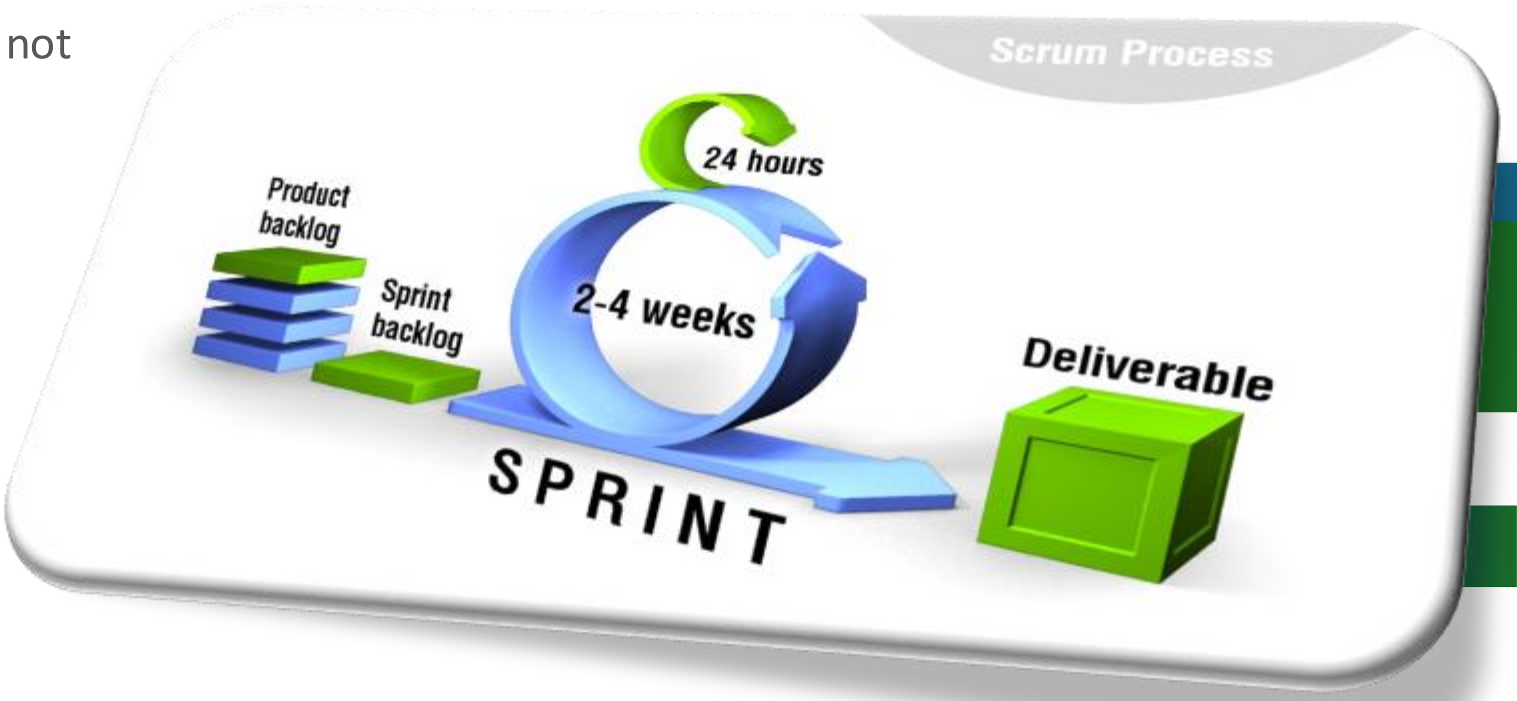
the creators of the Scrum method believe that no matter how much you plan, as soon as the software begins to be developed, chaos breaks out and the plans go out the window.

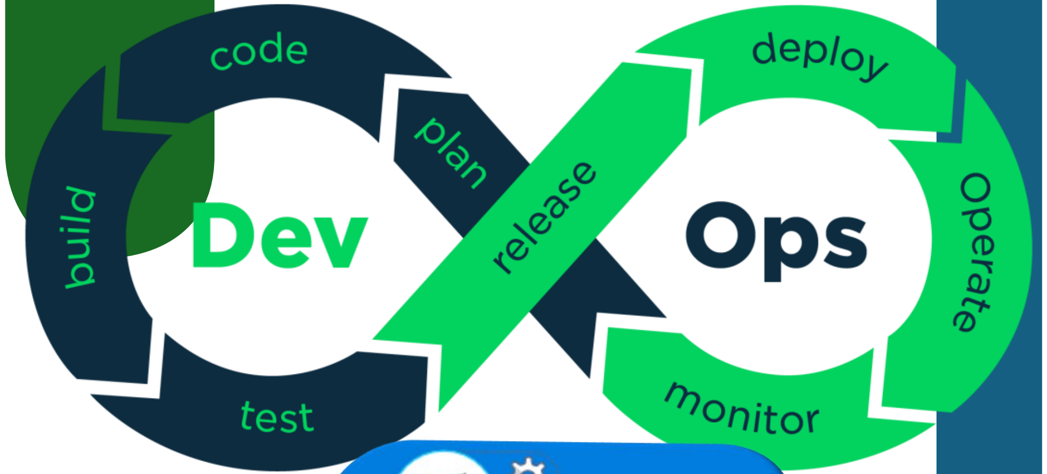
Teams are self-organized and self-directed. Unlike other approaches, Scrum teams do not have a designated team leader.

SCRUM Method

- Teams organize themselves in a symbiotic manner and set their own goals for each sprint (iteration).

- Once a sprint has begun, Scrum teams do not consider any additional requirements.



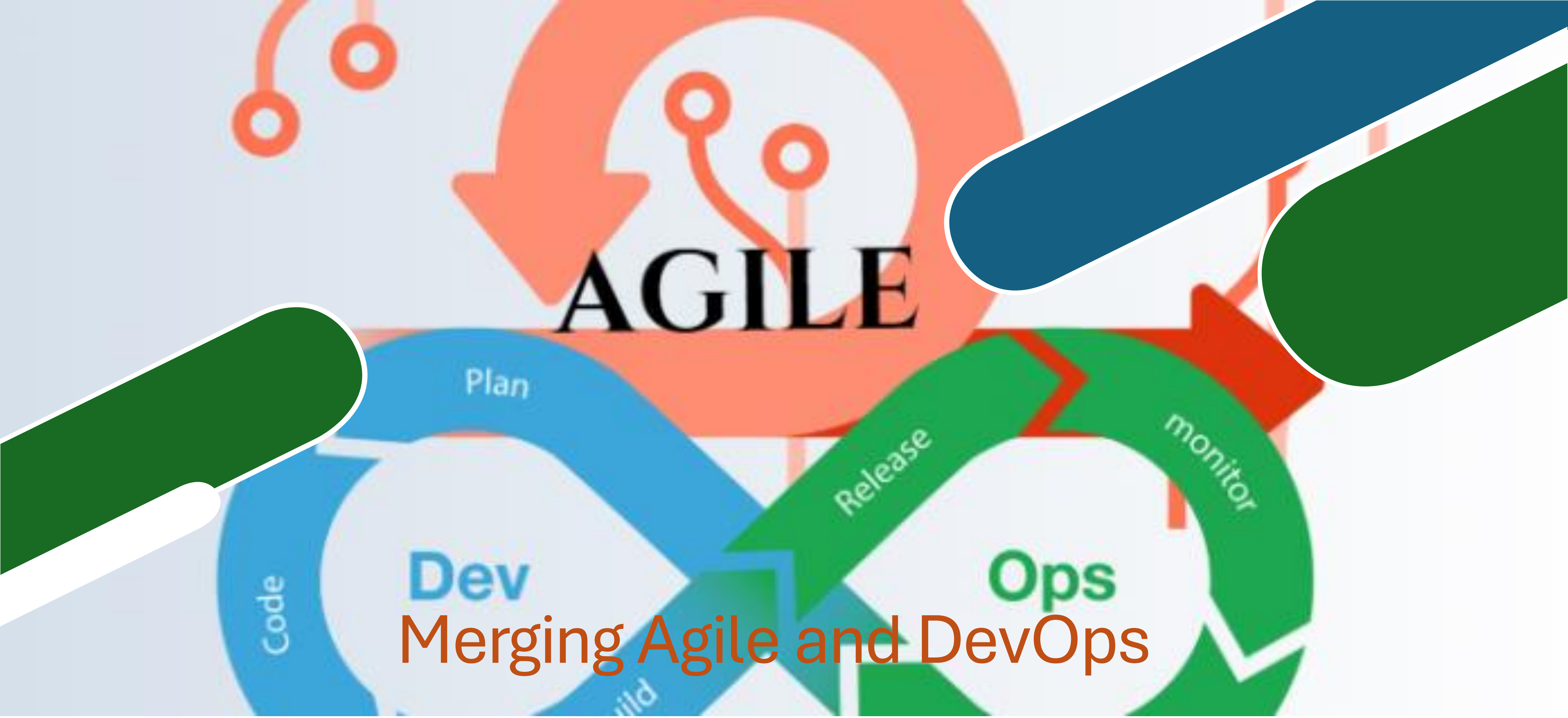


Microsoft Azure DevOps provides developer services to support teams to plan work, collaborate on code development, and build and deploy applications.

DevOps

DevOps is a set of practices that combines software development and IT operations. It aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

DevOps is complementary with Agile software development; several DevOps aspects came from Agile methodology.



Merging Agile and DevOps

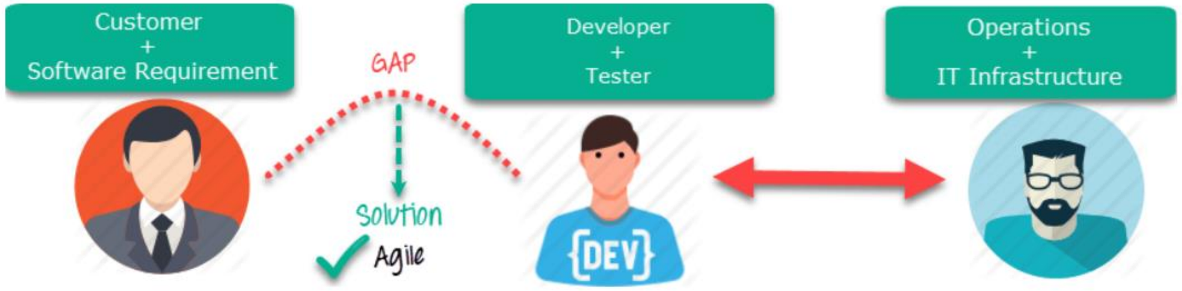
The Merging of DevOps and Agile approaches improve the quality and decrease the time of project for 37%.

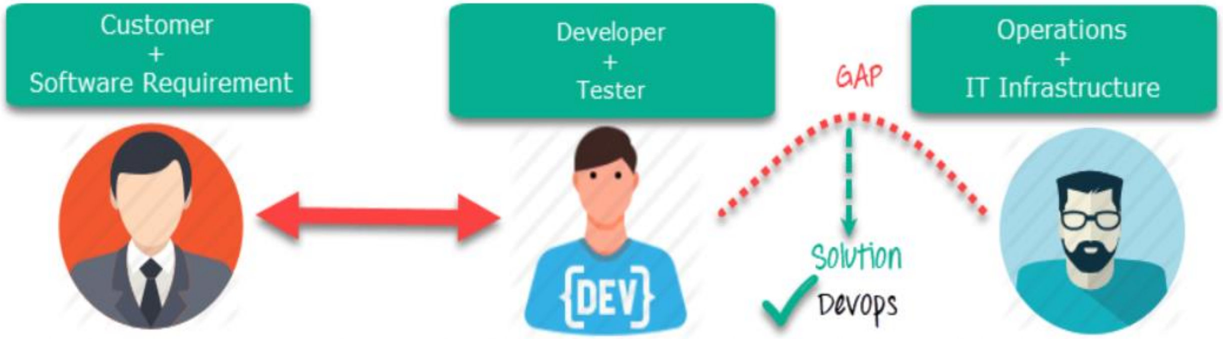
DevOps And AGILE

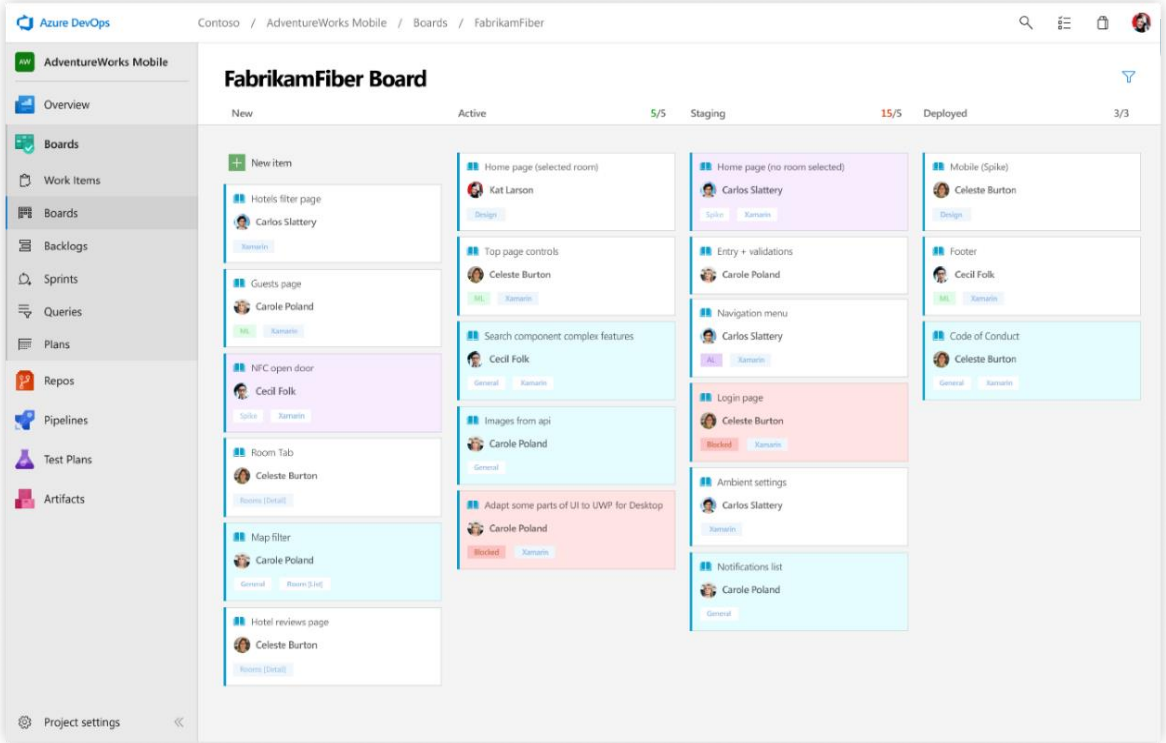
Agile improve the communication between the customer with development team

DevOps can address the gap between development and operation team within a cloud environment .

All Cloud DevOps Environments now merge the 2 methods to solve the problem Like Azure DevOps







FabrikamFiber Board

New

New item

Hotels filter page
Carlos Slattery
Xamarin

Guests page
Carole Poland
ML Xamarin

NFC open door
Cecil Folk
Spike Xamarin

Room Tab
Celeste Burton
Rooms (Detail)

Map filter
Carole Poland
General Room (List)

Hotel reviews page
Celeste Burton
Rooms (Detail)

Active 5/5

Home page (selected room)
Kat Larson
Design

Top page controls
Celeste Burton
ML Xamarin

Search component complex features
Cecil Folk
General Xamarin

Images from api
Carole Poland
General

Adapt some parts of UI to UWP for Desktop
Carole Poland
Blocked Xamarin

Staging 15/5

Home page (no room selected)
Carlos Slattery
Spike Xamarin

Entry + validations
Carole Poland

Navigation menu
Carlos Slattery
AL Xamarin

Login page
Celeste Burton
Blocked Xamarin

Ambient settings
Carlos Slattery
Xamarin

Notifications list
Carole Poland
General

Deployed 3/3

Mobile (Spike)
Celeste Burton
Design

Footer
Cecil Folk
ML Xamarin

Code of Conduct
Celeste Burton
General Xamarin

Criteria to select a Method

Choosing a methodology is not simple, because no one methodology is always best.

Criteria

Ability to Develop Systems	Structured Methodologies		RAD Methodologies		Agile Methodologies	
	waterfall	parallel	phased	prototype	XP	SCRUM
With Unclear User Requirements	poor	poor	Good	Excellent	Excellent	Excellent
With Unfamiliar Technology	poor	poor	Good	poor	Good	Good
That Are Complex	Good	Good	Good	poor	Good	Good
That Are Reliable	Good	Good	Good	poor	Excellent	Excellent
With a Short Time Schedule	poor	Good	Excellent	Good	Excellent	Excellent
With Schedule Visibility	poor	poor	Excellent	Good	Excellent	Excellent

THANK YOU!