

WorldVisitz

Mobile Application Agile Solution

1

Our Goals and Constraints

2

2.1 Agile Development Overview

2.2 Waterfall vs. Agile

3

The Optimal Agile Framework --- Scrum

Comparing Scrum, Kanban and XP

Goals

Product

1

Mobile App

1. Book travel with major airlines.
2. Predictive analysis functionality.
3. Additional services and offerings from other companies

Process

2

Development Process

1. Increase knowledge transfer.
2. More efficient development processes.
3. Shorten ROI (less than 18 months)

Constrains

- 1.Lack of knowledge transfer
- 2.No adhering coding standard
- 3.Lack of trust and camaraderie in the development team
- 4.Inefficient working procedures

Agile Manifesto - 4 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

Reliable

- Many Frameworks are developed
- Many organizations have practices of Agile development
 - Higher quality delivery
 - Higher level of predictivity

Waterfall

Linear fashion project.

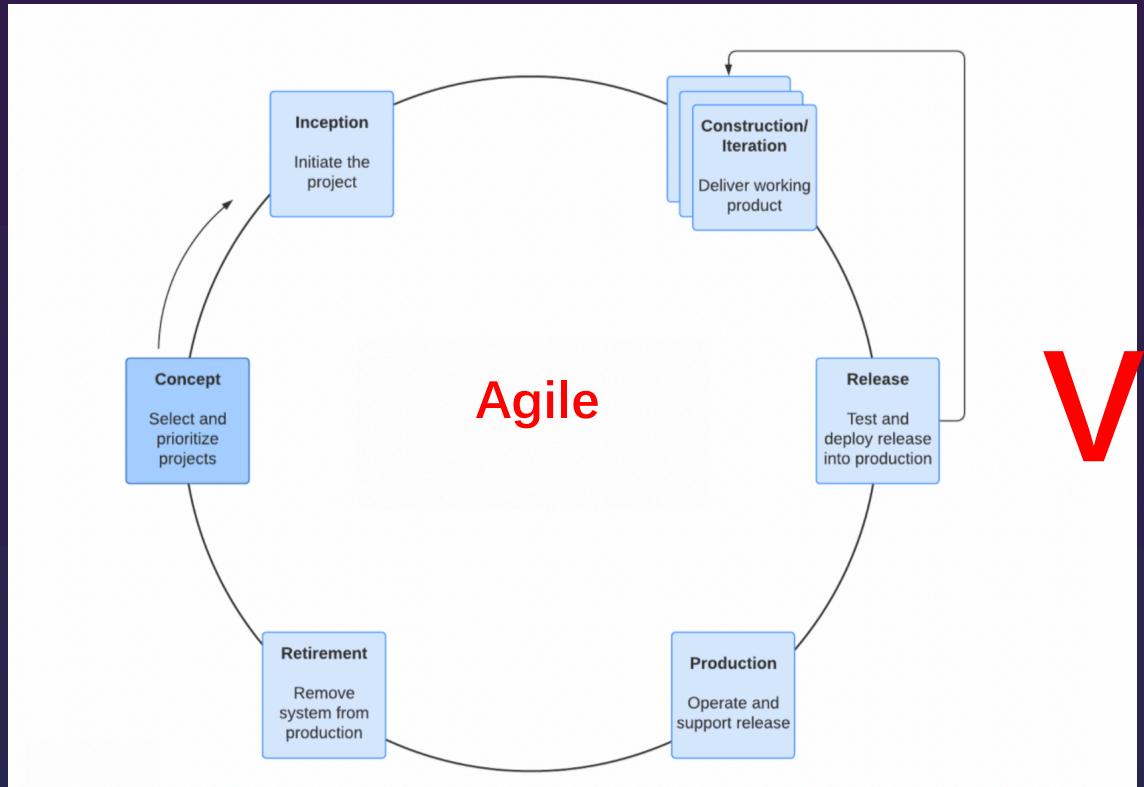
Do not need to go back to a prior phase.

Agile

Adaptive, simultaneous workflow.

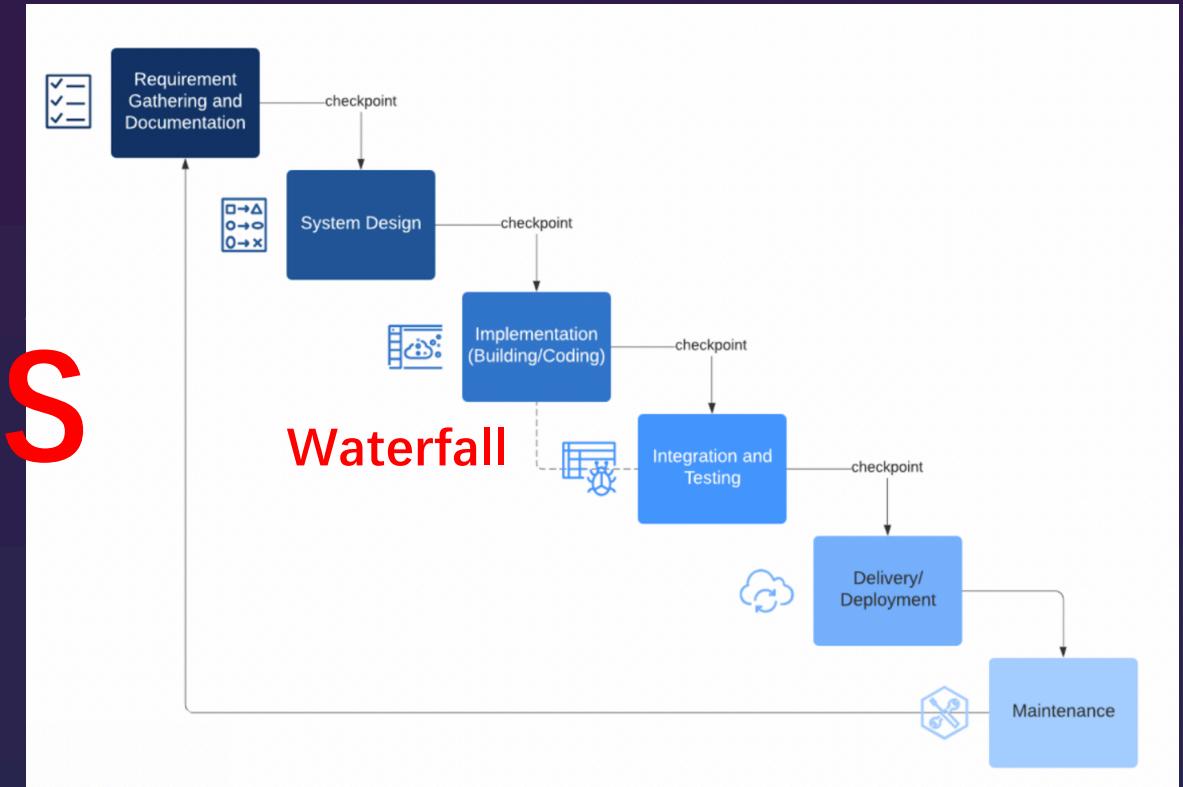
Break projects into smaller, iterative periods.

Agile Software Development Life Cycle Overview



Agile

Waterfall Software Development Life Cycle Overview



VS

1. Simultaneous, incremental work
2. Cross-functional team
3. Adaptability

1. One phase must be completed before another begins
2. Waterfall uses distinct phases rather than simultaneous work.
3. Documenting requirements

Agile is a more effective solution for this project and WorldVisitz

	Waterfall	Agile	Value
Product lifecycle	Distinct phases	Iterative	<i>Speed up the product lifecycle</i>
Design process	Sequential	Incremental	<i>Speed up the product lifecycle</i>
Change	Challenges	Embrace change	<i>Increase value</i>
Collaboration	Limited Synchronization	Significant collaboration	<i>Knowledge transfer</i>
Team	Homogeneous	Cross-function	<i>Knowledge transfer</i>
Requirements	Prepared at start of project	Prepared incrementally	<i>Increase value</i>
Documentation	Comprehensive documentation	Light-weight documentation	<i>Increase value</i>

Agile Umbrella

'Agile' is an umbrella term for all the methods and methodologies that follow the values and principles described in the Agile Manifesto.

Organizations can be agile by using any of the available Agile methodologies (Frameworks).



Agile Principles

1. **Customer satisfaction through early and continuous software delivery** – Customers are happier when they receive working software at regular intervals, rather than waiting extended periods of time between releases.
2. **Accommodate changing requirements throughout the development process** – The ability to avoid delays when a requirement or feature request changes.
3. **Frequent delivery of working software** – Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.
4. **Collaboration between the business stakeholders and developers throughout the project** – Better decisions are made when the business and technical team are aligned.
5. **Support, trust, and motivate the people involved** – Motivated teams are more likely to deliver their best work than unhappy teams.
6. **Enable face-to-face interactions** – Communication is more successful when development teams are co-located.
7. **Working software is the primary measure of progress** – Delivering functional software to the customer is the ultimate factor that measures progress.
8. **Agile processes to support a consistent development pace** – Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.
9. **Attention to technical detail and design enhances agility** – The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.
10. **Simplicity** – Develop just enough to get the job done for right now.
11. **Self-organizing teams encourage great architectures, requirements, and designs** – Skilled and motivated team members who have decision-making power, take ownership, communicate regularly with other team members, and share ideas that deliver quality products.
12. **Regular reflections on how to become more effective** – Self-improvement, process improvement, advancing skills, and techniques help team members work more efficiently

Three Agile Frameworks

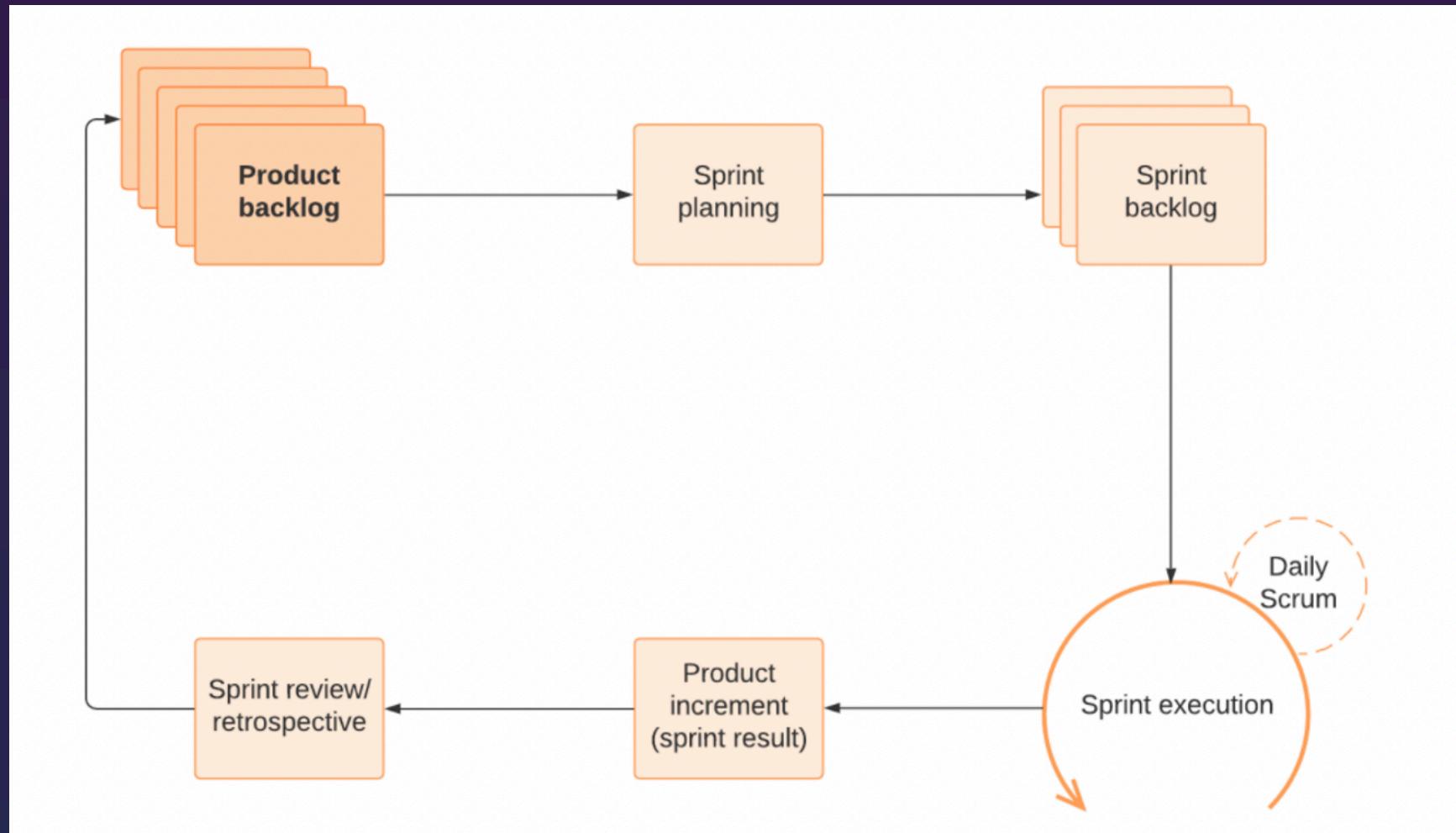
Comparing the three agile frameworks

Scrum

- The features are delivered in iterations known as Sprints.
- The Team conducts Daily Stand-Ups in which they provide updates to each other.
- Retrospectives are conducted to identify opportunities for improvement at the end of each Sprint.
- Specific roles, as well as certain Ceremonies are recommended in the Scrum Framework.

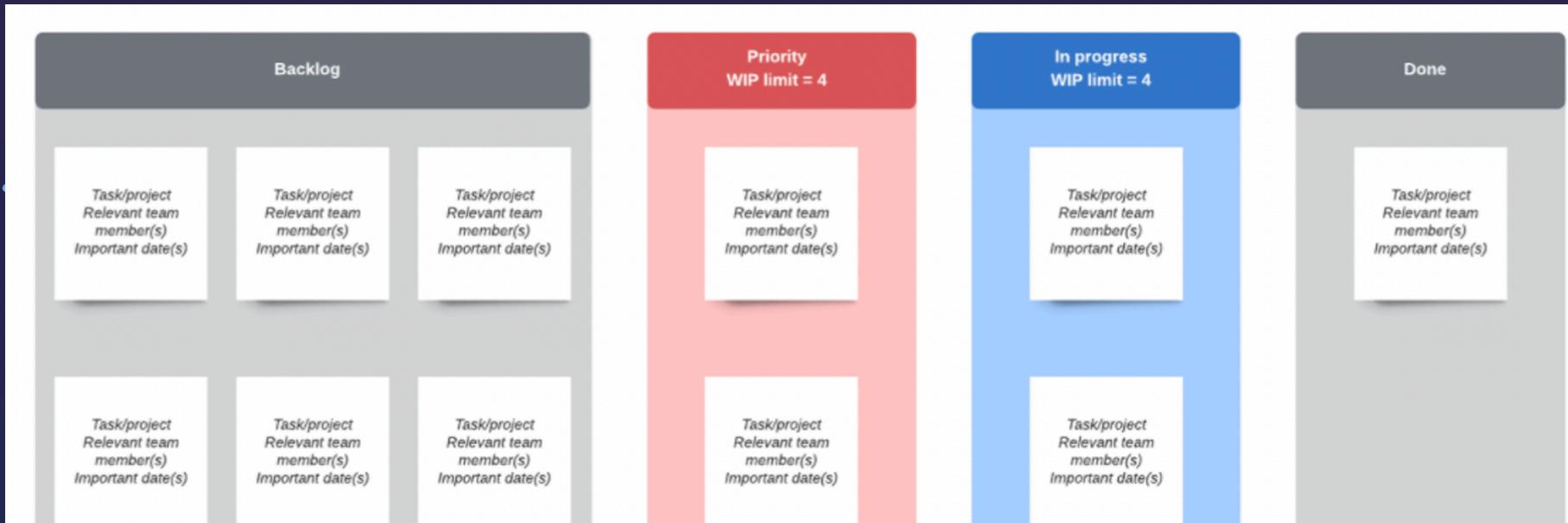
Iterations

Scrum SDLC



Kanban Board

- Workflow is visualized in Kanban boards.
- Continuous Improvement.
- Backlog, Work in Progress(WIP), Done



XP

- XP – Extreme Programming
- This agile framework with a strong focus on technical excellent and best practices in software development.
- XP tends to deliver high quality software as it emphasizes software system design, coding quality and testing.

Comparison

- **XP** is more towards Engineering process. It brings a strong focus on quality by insisting on a set of core engineering practices including testing, code refactoring and so on.
- **Kanban** is a useful tool for teams with a continually changing backlog of items to increase efficiency by limiting the amount of work-in-progress, and the workflow is visualized.
- **Scrum** builds cross-functional, self-organized teams that members cooperate to deliver significant value quickly cross projects.

Our Choice - Scrum

Challenges

- Product/Process
- 1. Complex functionalities
 - 2. Integrate with external companies services
 - 3. Less than 18 months
 - 4. Inefficient working procedures
 - 5. No coding standard
 - 6. Poor teamwork

XP

Pros

- 1. Code quality, Software stable
- 2. Coding standard
- 3. Communication
- 4. Procedures

Con.

- Time
- Experienced developers

Scrum

Pros

- 1. Collaboration
- 2. Short work cycle
- 3. Iterations
- 4. Knowledge transfer

Con.

- Code quality

Scaling Agile

1. MVP

- A minimum viable product (MVP) is a concept from Lean Startup that stresses the impact of learning in new product development. The new version of a product which allows a team to collect the maximum amount of validated learning about customers with the least effort.
- As WorldVisitz is going to develop this complex application, we can employ MVP in the development process. Gathering customer feedback constantly; followed up by iterations, to improve customer satisfaction and software quality.

2. Cultivate a Culture of Collaboration

- Cross-functional teams are invited to attend Sprint Demo to provide feedback.
- Cross-functional teams should join the Architecture design meeting to propose requirements in the early stage (avoid rework/waste).
- The key is to create a safe environment that everyone is willing to work with others and speak their real thoughts.

3. Design a Coding Manual

- Since WorldVisitz does not have an adhering coding standard, we can make a catch-up inviting the development team and tester to make a coding manual that sets a coding standard.