Agile Methodologies

What is Agile Development?

- Agile Development is an umbrella term for several iterative and incremental software development methodologies.

  Samples for these methodologies include: Scrum, Lean Development, Extreme Programming and others.

- Although each one of these methodologies is unique and different than the rest, these methodologies share a common vision and core values.

The Agile Manifesto

"We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value the following:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan"

(http://agilemanifesto.org/)

Agile Principles

- ## Active Involvement

  All project members, the project owner and the customer (or a representative) should be actively involved in the project.

- ## Empowering Agile Development Team

  Empowering the development team to make decisions will ensure their understanding of their responsibility to deliver the product on time. Empowering the development team will create the required ownership feeling.

- ## Capture Requirements at High Level

  This way we shall minimize time spent on anything that is not part of the final project.

Agile Principles

❖ Rigid Time Frame

Unlike other methodologies, when using an agile methodology it is clear that changes can be introduced during the development. Changes are acceptable. Yet, in order to keep us within the budget frame the rigid time frame should be clear to everyone. This way, it would be clear that introducing a big change might cause taking out other another part or parts of the project.

❖ Small Incremental Releases

Developing small incremental releases and iterate. While in a non-agile methodology we usually one cycle that include the analyze, develop and test phases for the whole project, when using an agile methodology we will repeat this cycle for each feature separately.

Agile Principles

❖ Frequent Delivery of Releases

There is a need to choose what is best for the project we develop and be careful of being too frequent. Projects on the web are a great example for the frequent deliveries. As soon as we can let our site to go live and deliver some value we will do so. We can always state it is a beta version. We won't wait for completing all planned features.

❖ Don't Leave Unfinished Feature

When using the agile methodology “Done” means “Done”. Don't leave unfinished features. Each feature should be completed within its iteration. You should complete each feature before moving forward to the next feature.

Agile Principles

- ## Apply The 80/20 Rule

  The Pareto's law, 80% of the results may come from 20% of the efforts, apply in software development as well. Try to focus on the 20%.

- ## Testing is Integrated Throughout The Life Cycle

  Write separated automated tests units for each feature. Test each feature as soon as you complete it. This way you will validate your code and if there is a need in any code changes you will introduce them in “no time” as the code is still fresh in your head. Make sure to incorporate within the cycle tests performed by potential product users as well. The integrated testing is crucial for delivering small incremental releases.

Agile Principles

- ## Close Collaboration

  Close collaboration and cooperation between all team members and the stake holders is crucial. This way, requirements will be clarified just-in-time and everyone will be kept on the same page.

Agile Advantages

- ## Software Quality

  The testing is integrated throughout the life cycle. Doing so enables regular inspection of the developed product. Having a team work together with the project owner and/or the user while being flexible to introduce new required changes (based on actual using and seeing the product) ensures the customers satisfaction, which yields to a better product.

- ## Higher Revenues

  New features of the developed software are delivered incrementally. That enables releasing the new features earlier than it would have been possible if a non Agile method was in use. That leads to higher revenues.

Agile Advantages

- ## Changes Flexibility

  The testing is integrated throughout the life cycle. Unlike other methodologies, when using an Agile methodology we are flexible. Changes are expected and welcomed. Requested changes don't create severe administrative complexities.

- ## The Right Product

  The emphasize is on getting the right product. Carefully and accurately following the specification. Trying to get the required product in our first try.

- ## More Enjoyable

  The active involvement and the cooperation with others turn the agile methodology into a more enjoyable for most developers.

Agile Advantages

- **Better Budget Control**

  The transparency we get when using an agile methodology and the fact the product is continuously developed improves our control over the total cost.

- **Better Risks Management**

  The transparency we enjoy when using an agile methodology (we can see what was already developed at each stage...) and the fact the product is continuously developed while having the management involved enables getting necessary decisions at the earliest possible opportunity. If, as in many software projects, we run out of budget or a crisis in the company forces us to put the project on hold, some value can still be realized.

Agile Advantages

- ## Speed to Market

  Agile methodologies enable us to get our system up and running faster than any other methodology.

- ## Market Opportunities

  Agile methodologies enable us to respond faster than any other methodology. Being fast and flexible we can add features and introduce required changes exploiting market opportunities faster than any other methodology.

- ## Higher Value

  Having the product delivered via many subsequent releases yields immediate value as soon as the first release is ready. The next release will be based on consumers feedback. All that faster than using any other methodology.

Agile Methodologies

What is Agile Development?

- Agile Development is an umbrella term for several iterative and incremental software development methodologies.
  Samples for these methodologies include: Scrum, Lean Development, Extreme Programming and others.
- Although each one of these methodologies is unique and different than the rest, these methodologies share a common vision and core values.

The Agile Manifesto

"We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value the following:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan"

(http://agilemanifesto.org/)

Agile Principles

- **Active Involvement**

  All project members, the project owner and the customer (or a representative) should be actively involved in the project.

- **Empowering Agile Development Team**

  Empowering the development team to make decisions will ensure their understanding of their responsibility to deliver the product on time. Empowering the development team will create the required ownership feeling.

- **Capture Requirements at High Level**

  This way we shall minimize time spent on anything that is not part of the final project.

Agile Principles

- **Rigid Time Frame**

  Unlike other methodologies, when using an agile methodology it is clear that changes can be introduced during the development. Changes are acceptable. Yet, in order to keep us within the budget frame the rigid time frame should be clear to everyone. This way, it would be clear that introducing a big change might cause taking out other another part or parts of the project.

- **Small Incremental Releases**

  Developing small incremental releases and iterate. While in a non-agile methodology we usually one cycle that include the analyze, develop and test phases for the whole project, when using an agile methodology we will repeat this cycle for each feature separately.

Agile Principles

- **Frequent Delivery of Releases**

  There is a need to choose what is best for the project we develop and be careful of being too frequent. Projects on the web are a great example for the frequent deliveries. As soon as we can let our site to go live and deliver some value we will do so. We can always state it is a beta version. We won't wait for completing all planned features.

- **Don't Leave Unfinished Feature**

  When using the agile methodology “Done” means “Done”. Don't leave unfinished features. Each feature should be completed within its iteration. You should complete each feature before moving forward to the next feature.

Agile Principles

- **Apply The 80/20 Rule**

  The Pareto's law, 80% of the results may come from 20% of the efforts, apply in software development as well. Try to focus on the 20%.

- **Testing is Integrated Throughout The Life Cycle**

  Write separated automated tests units for each feature. Test each feature as soon as you complete it. This way you will validate your code and if there is a need in any code changes you will introduce them in “no time” as the code is still fresh in your head. Make sure to incorporate within the cycle tests performed by potential product users as well. The integrated testing is crucial for delivering small incremental releases.

Agile Principles

- **Close Collaboration**

  Close collaboration and cooperation between all team members and the stake holders is crucial. This way, requirements will be clarified just-in-time and everyone will be kept on the same page.

Agile Advantages

- **Software Quality**

  The testing is integrated throughout the life cycle. Doing so enables regular inspection of the developed product. Having a team work together with the project owner and/or the user while being flexible to introduce new required changes (based on actual using and seeing the product) ensures the customers satisfaction, which yields to a better product.

- **Higher Revenues**

  New features of the developed software are delivered incrementally. That enables releasing the new features earlier than it would have been possible if a non Agile method was in use. That leads to higher revenues.

Agile Advantages

- **Changes Flexibility**

  The testing is integrated throughout the life cycle. Unlike other methodologies, when using an Agile methodology we are flexible. Changes are expected and welcomed. Requested changes don't create severe administrative complexities.

- **The Right Product**

  The emphasize is on getting the right product. Carefully and accurately following the specification. Trying to get the required product in our first try.

- **More Enjoyable**

  The active involvement and the cooperation with others turn the agile methodology into a more enjoyable for most developers.

Agile Advantages

- **Better Budget Control**

  The transparency we get when using an agile methodology and the fact the product is continuously developed improves our control over the total cost.

- **Better Risks Management**

  The transparency we enjoy when using an agile methodology (we can see what was already developed at each stage...) and the fact the product is continuously developed while having the management involved enables getting necessary decisions at the earliest possible opportunity. If, as in many software projects, we run out of budget or a crisis in the company forces us to put the project on hold, some value can still be realized.

Agile Advantages

- **Speed to Market**

  Agile methodologies enable us to get our system up and running faster than any other methodology.

- **Market Opportunities**

  Agile methodologies enable us to respond faster than any other methodology. Being fast and flexible we can add features and introduce required changes exploiting market opportunities faster than any other methodology.

- **Higher Value**

  Having the product delivered via many subsequent releases yields immediate value as soon as the first release is ready. The next release will be based on consumers feedback. All that faster than using any other methodology.