Agile Introduction For Dummies

Agile Transformations translate to radical visibility!

WATERFALL vs. AGILE METHODOLOGY

There is no IT meeting that does not talk and debate endlessly about Waterfall vs. Agile development methodologies. Feelings run strong on the subject with many considering **Agile** *'so of the moment',* just so right, while **Waterfall** is thought to be passé! But, before deciding which is more appropriate, it is essentially important to provide a little background on both.

Waterfall

A classically linear and sequential approach to software design and systems development, each waterfall stage is assigned to a separate team to ensure greater project and deadline control, important for on-time project delivery. A linear approach means a stage by stage approach for product building, e.g.

1. The project team first **analyses**, then determining and prioritising business requirements / needs.

2. Next, in the **design phase** business requirements are translated into IT solutions, and a decision taken about which underlying technology i.e. COBOL, Java or Visual Basic, etc. etc. is to be used.

3. Once processes are defined and online layouts built, code **implementation** takes place.

4. The next stage of data conversion evolves into a fully **tested** solution for implementation and testing for evaluation by the end-user.

5. The last and final stage involves **evaluation** and **maintenance,** with the latter ensuring everything runs smoothly.

However, in case a glitch should result, changing the software is not only a practical impossibility, but means one has to go right back to the beginning and start developing new code, all over again. That's Waterfall for you! Now, as for minimal risk Agile, it is a low over-head method that emphasizes values and principles rather than processes. Working in cycles i.e. a week, a month, etc., project priorities are re-evaluated and at the end of each cycle. Four principles that constitute Agile methods are:

1. The reigning supreme of individuals and interactions over processes and tools.

2. As does, working software over comprehensive documentation.

3. Likewise, customer collaboration over contract negotiation.

4. And again, responding to change over plan follow-throughs.

To synopsise the difference between the two, one can say the classic waterfall method stands for predictability, while Agile methodology spells adaptability. Agile methods are good at reducing overheads, such as, rationale, justification, documentation and meetings, keeping them as low as is possible. And, that is why Agile methods benefit small teams with constantly changing requirements, rather more than larger projects.

Agile, based on empirical rather than defined methods (Waterfall) is all about light maneuverability and sufficiency for facilitating future development. By defined methods what one means is that one plans first and then enforces these plans. However, Agile methods involve planning what one wants and then adapting these plans to the results. Extreme Programming (XP) is an excellent example of Agile methodology i.e.:

1. Communication between customers and other team members;

2. Simple, clean designs;

3. Feedback given on Day 1 of software testing;

4. Early delivery and implementation of suggested changes.

Agile methodology means cutting down the big picture into puzzle size bits, fitting them together when the time is right e.g. design, coding and testing bits. So, while there are reasons to support both the waterfall and agile methods, however, a closer look clarifies why many software and web design firms make the more appropriate choice of employing Agile methodology. The following table enumerates the raison d'être for choosing Agile methodology over the Waterfall method.

1. Once a stage is completed in the **Waterfall method,** there is no going back, since most software designed and implemented under the waterfall method is hard to change according to time and user needs. The problem can only be fixed by going back and designing an entirely new system, a very costly and inefficient method. Whereas, **Agile methods** adapt to change, as at the end of each stage, the logical programme, designed to cope and adapt to new ideas from the outset, allows changes to be made easily. With Agile, changes can be made if necessary without getting the entire programme rewritten. This approach not only reduces overheads, it also helps in the upgrading of programmes.

2. Another **Agile method** advantage is one has a launchable product at the end of each tested stage. This ensures bugs are caught and eliminated in the development cycle, and the product is double tested again after the first bug elimination. This is not possible for the **Waterfall method,** since the product is tested only at the very end, which means any bugs found results in the entire programme having to be re-written.

3. **Agile's** modular nature means employing better suited object-oriented designs and programmes, which means one always has a working model for timely release even when it does not always entirely match customer specifications. Whereas, there is only one main release in the waterfall method and any problems or delays mean highly dissatisfied customers.

4. Agile methods allow for specification changes as per end-user's requirements, spelling customer satisfaction. As already mentioned, this is not possible when the waterfall method is employed, since any changes to be made means the project has to be started all over again.

5. However, both methods do allow for a sort of departmentalization e.g. in waterfall departmentalization is done at each stage. As for Agile, each coding module can be delegated to separate groups. This allows for several parts of the project to be done at the same time, though departmentalization is more effectively used in Agile methodologies.

In conclusion, though on the plus side, waterfall's defined stages allow for thorough planning, especially for logical design, implementation and deployment, Agile methodology is a sound choice for software development and web design projects. More and more firms are becoming Agile!