AGILE METHODOLOGY

The agile methodology is an iterative (that is, repetitive) model, that comprises a cyclic and collaborative process. With this method, software is developed in incremental, rapid cycles. This results in additive releases, in which each release builds on the previous functionality to make improvements where necessary. This method is used for time-critical applications. It is also used for projects where the scope and timeline for the development of the software may be difficult to ascertain in advance and so the project is planned in stages or "sprints" (as they are commonly called), which evolve as the work progresses. As such, this model is suitable for urgent projects or projects with a fair amount of uncertainty.

PROS OF THE AGILE METHODOLOGY

1. It is a people-oriented process. That is, customers, developers and testers constantly interact with each other as their input is required at every stage.
2. There is customer satisfaction based on the rapid and continuous delivery of useful software.
3. Working software is delivered within a short time. Hence, the product gets to the market faster.
4. There is regular adaptation to changing circumstances.
5. Large changes to requirements are welcomed.

CONS OF AGILE METHODOLOGY

1. Poor resource planning.
2. Documentation of processes is minimal.
3. Output is fragmented.
4. Processes are repetitive.
5. Final product is not released first.
6. No accurate estimation of project cost.

WATERFALL METHODOLOGY

The waterfall methodology is a sequential methodology in which tasks are handled in a more linear process, where each phase depends on the deliverables of the previous one. There is also specialization of task in each phase. Strict standards and requirements are followed using this model and as such is suitable for high-risk projects.

PROS OF WATERFALL METHODOLOGY

1. It is simple and easy to use and understand.
2. It is easy to manage due to the rigidity of the model as each phase has specific deliverables and a review process.
3. Phases are well defined and completed one at a time.
4. There is proper documentation and so developers who join the project can get up to speed as the system's requirements are in the requirements document.
5. Project cost can be accurately estimated.

CONS OF WATERFALL METHODOLOGY

1. No working software is produced until late during the life cycle.
2. It is difficult to make changes at other stages of the lifecycle other than at the requirements stage which marks the onset of the project.
3. One cannot move onto the next stage of development unless they complete the previous one, which may make the whole process run for a longer time.
4. There is a high amount of risk and uncertainty because one is unable to tell the outcome of a product until the end of all stages. Testing is delayed and this could be costly.
5. It excludes the customers and end users. As a result, stakeholders are less informed throughout the life of the project.