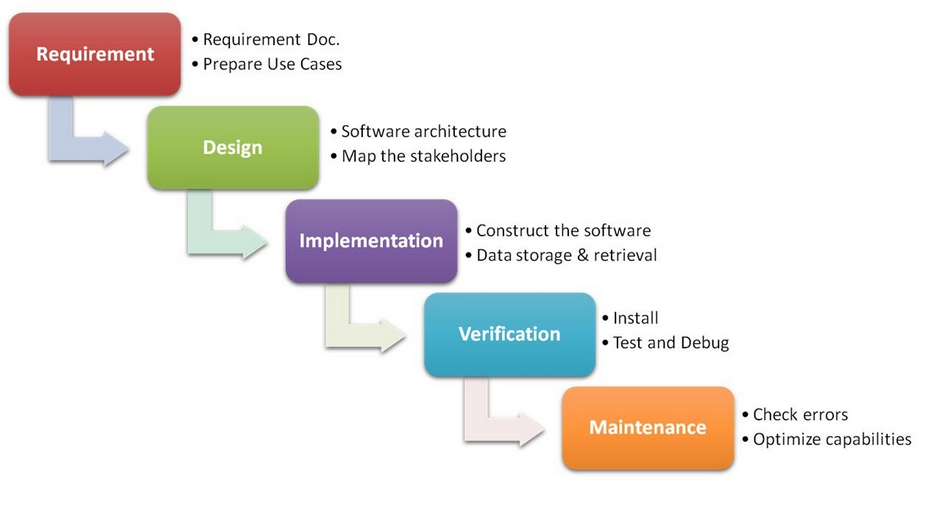
# **Evaluation of Software Development Methodologies**

**Possible Software development Methodologies:**

A software development methodology is a framework used to structure and manage the overall development of a software development project and the final system or solution. My project is to create a helpdesk system for Numatic, following a development methodology will ensure all the required parts of the lifecycle are complete and will assist in the management of the project. Whilst there are many different methodologies, below I have investigated and discussed various different methods, this will allow me to select the most appropriate and beneficial method for this project.

**Waterfall Development:**

This is one of the most common software development methodologies used when developing software. The Waterfall model stipulates that each stage is required and must be followed in sequence. Additionally each stage is completed by either validation, verification or testing to make sure the stage has been completed successfully and requires an agreement to sign off on a phase and move onto the next stage.



The main emphasis of this methodology is the emphasis that is placed on planning, strict deadlines and short time span; in those instances the Waterfall model works well to minimise process overhead. Following the Waterfall model yields a well-structured and controlled approach, with tight control over the whole project down to the extensive documentation. These standards ensure nothing is missing from the specifications before progressing from one stage onto the next stage and allows the client to sign off each stage. Progress is reviewed at the end of each stage thus helping to achieve specified dates in the project plan, the emphasis on documentation and design before building the system helps minimize time and effort wastage.

When using the Waterfall model, going back to any previous stage should be avoided at all costs, therefore it is most important to correctly identify all requirements up front. All system development is completed in a single stage, so the customer will not see many signs of progress until nearing the end of the project, thus projects should not be time critical.

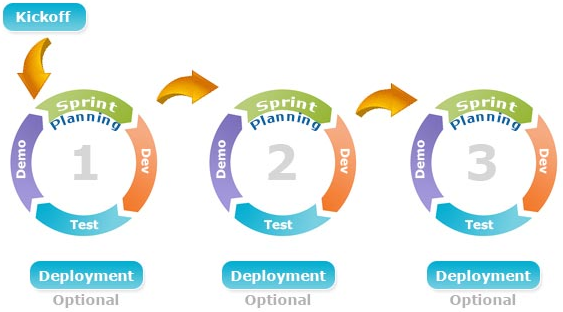
The main disadvantage with this methodology is that the requirements have to be very well defined and the client has to be sure they will not want to add additional requirements after the requirement stage has been completed. If this happens it will have a knock on effect throughout the project, meaning a delay and this methodology does not have a way of controlling this scenario.

There are some advantages of using this methodology, the model is very simple and easy to use, allowing inexperienced staff and managers to manage a project effectively. Having defined stages allows the progress of the project to be measured and the strict controls over stages ensure the products from each stage are of high quality and that the client is happy with them.

In projects where the requirements are formally set out and will not change during the project, this methodology is most suitable. Also projects that are not time critical, this is due to the fact that as there is only one iteration of the lifecycle if it is late then there will be no system in place.

**Agile Approach**

Unlike the Waterfall approach where the requirements are fully defined and signed off before development begins, the iterative approach begins by specifying and developing a certain part of the software/solution. At the end of the iteration the software is reviewed and helps to identify any further requirements. This is then repeated until the client agree they are happy, there are many different agile practises, such as, extreme programming and scrum.



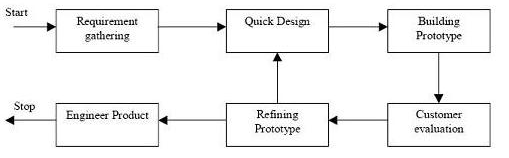
The rapidity of development is one of main advantages of choosing an agile approach, having a minimum level of a working system quickly is often much preferable than having to wait much longer for a complete system. The high levels of communication and less rigid approach allows for much more freedom and consequently leads to an end product that the end users want.

However, there are some disadvantages of agile, heavy client participation is required which can be costly and time consuming. There is also a large possibility of scope and cost to increase during the project as new requirements emerge and evolve.

Agile is best implemented with projects that involve new systems, where the requirements are not set in stone and may need changing, changes can be changed quickly due to the iteration process of development.

**Prototyping Approach**

Prototyping is a useful software development methodology allowing, prototypes to be developed for a particular functionality or parts of the system. Prototypes are built to allow the client to look at a system which has the current requirements, this allows them to get a feel of the system and to help with requirement elicitation.



The main advantage of this methodology is the client and user interaction during the project, meaning they will have a clear understanding of the system being developed; enabling feedback from the client to be quick. This high level of interaction increases the quality of the end product and allows for the developers to also ensure they fully understand the requirements and missing functionality can be easily identified.

There also disadvantages of this methodology, such as, there being too much dependency on prototypes and the prototypes being used in the development of the actual system even if it is not appropriate. The possibility of confusing the end users between prototypes and the end product may also happen.

Prototyping is most successful when the project requires a high level of end user interaction, it also allows the users and client to be continually working with the system as it is developed. This is normally best for large projects or projects with complex requirements.

**Conclusion of Development Methodologies:**

After investigating many different methods of software development, I have concluded the agile model is the most relevant to my project. Whilst all software development methods have their own strengths and weaknesses, the agile approach fits in well with the project. Separating the project into stages allows the client to quickly get a system and can see the product developing, perhaps more importantly the requirements can also be refined and developed throughout the project ensuring the client will be happy with the end product which should also increase the quality of the system.

References:

<http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/>

<http://istqbexamcertification.com/what-is-prototype-model-advantages-disadvantages-and-when-to-use-it/>

<http://www.tutorialspoint.com/sdlc/sdlc_software_prototyping.htm>

<http://www.iotap.com/blog/entryid/124/advantages-disadvantage-of-prototyping-process-model>

<http://www.allaboutagile.com/disadvantages-of-agile-development/>

<https://www.cms.gov/research-statistics-data-and-systems/cms-information-technology/xlc/downloads/selectingdevelopmentapproach.pdf>