

Exercise #3.2

SOEN6841

Question: Agile projects may have less effort required compared to traditional projects. What factors are responsible for this phenomenon?

Answer To the best of my knowledge after reading Chapter 3 and doing research online, the underlying reasons are categorized as follows:

1- Iterative Development & Early Feedback: The agile approach usually follows an incremental and iterative approach so that the project is built in small cycles instead of a single large release at the start of the project in the traditional format. Moreover, by receiving frequent feedback from stakeholders and end-users, we minimize the errors and misalignments that are corrected before they become costly. So, it increases the chances of early error detection and reduces the effort that might be required and vice versa. In traditional waterfall projects, major changes require revisiting entire project phases while we do less rework in agile projects as they allow us quick adjustments to reduce unnecessary effort.

2- Better Scope & Requirement Management: As I mentioned previously, Agile allows changing requirements and reducing the risk of wasted effort on unnecessary or outdated features. This leads to adaptive planning for Agile projects. Another aspect of such Agile projects is prioritizing features based on higher values they proposed to reduce effort on low-priority or unneeded functionalities.

3- Efficient Team Collaboration & Productivity: Traditional models require fixed team allocation and strict project plans, which can lead to underutilization of resources. Since it is explicitly mentioned in 3.2.5 that "traditional development model have to plan for everything in advance including making elaborate effort estimates and this cannot be done without proper identification of project tasks that will be involved". Therefore, these fixed upfront estimates will lead to inaccuracy because they are highly dependent on initial specifications, which might change. On the other hand, Agile adapts team workloads dynamically, leading to better productivity and optimized effort. This is well explained in 3.2.6 that "Agile model is best suited for projects where the risk of software development is very high similar to R&D projects with high degree of innovation and creativity. Instead of a fixed price/fixed duration model, a **time-and-material-based contract** suits such projects." So, If the project outcome is not encouraging, it can be abandoned early, reducing losses. Also, breaking the project into small, manageable iterations lowers financial and effort risks we'll have. Plus, In subsequent iterations, since the team has gained experience, estimates will improve over time based on real progress.

More interestingly, the book mentioned that the volume of work does not much exceed that of any waterfall model-based project (maybe 1/20 or even less) Due to these reasons, for iteration-based projects, effort and cost estimates are not a major concern for customers and thus, not a critical element of project management.

Reference image of factors impacting agile approach page 49

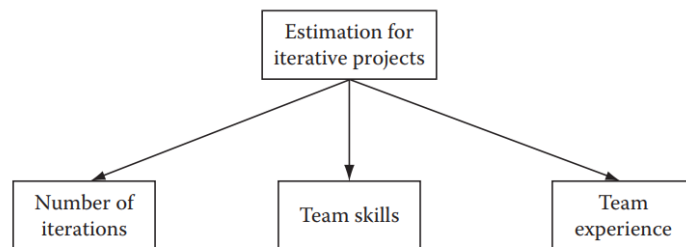
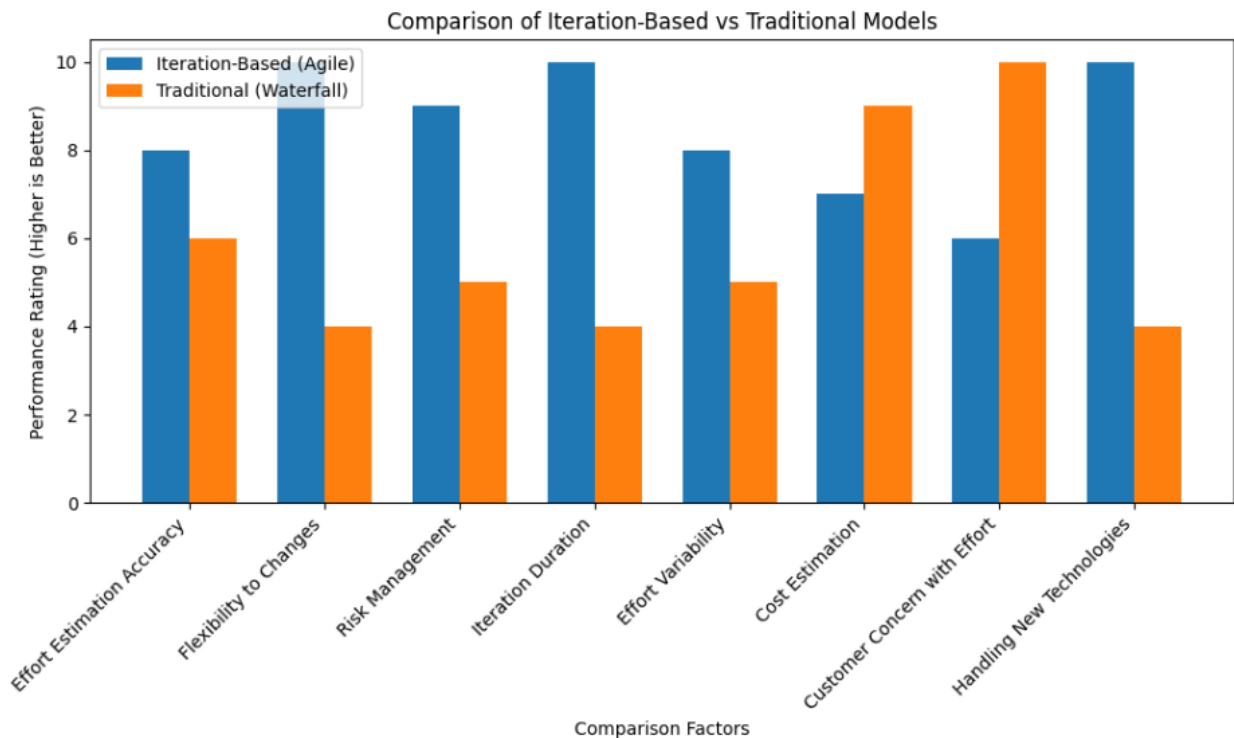


Figure 3.6 Factors influencing effort estimate for agile and iteration-based projects.

I also provide a summarised comparison table mostly as a note for a quick recall before the midterm.

Aspect	Iteration-Based Models (Agile, Iterative) <i>(Section 3.2.6)</i>	Traditional Models (Waterfall, RUP) <i>(Section 3.2.5)</i>
Accuracy	Effort estimation is uncertain due to initial estimates may be changing but improve over time and iterations.	The effort is estimated upfront; accuracy depends on initial planning, which may not capture real requirements.
Flexibility to Changes	Highly flexible; changes can be merged after each iteration without major rework.	Fixed; any change in requirements requires major rework and additional effort because of the sequential nature of the development process.
Risk Management	Lower risk as projects are divided into smaller, manageable iterations and minimized failure impact.	Higher risk; failures discovered late in the project that can be costly.
Iteration Duration	Short cycles of 2–8 weeks; quicker feedback enables faster adaptation.	With longer phases involved, feedback may take months or years to be received.
Effort Variability	The scope changes with each iteration, therefore each iteration is unique.	Fixed once the project plan is made; changes require a full reevaluation.
Customer Concern	Less important; customers accept that effort and cost will change over time.	Customers need precise cost and effort estimates before project execution.
Handling New Technologies	Ideal for new and emerging technologies that involve a significant degree of uncertainty like R&D projects.	Unsuitable for emerging technologies due to rigid planning and high uncertainty.

I made this comparison factor chart based on the general ideas I learned. Please note that **performance ratings (out of 10)** are based on my assumptions and imaginations and are not real numbers with reference!



Link to github: <https://github.com/HaniMLK/Soen-6841-/wiki/Exercise-3.2>

Key Takeaways

- Agile/Iteration-based models are better for projects with uncertainty, evolving requirements, and rapid innovation.
- Traditional Models (Waterfall/RUP) are better for projects with fixed requirements, well-defined scope, and low uncertainty.
- Agile requires less effort because it reduces rework, manages risk iteratively, and adjusts estimates dynamically.
- Waterfall projects demand more effort upfront due to extensive planning, rigid structures, and high rework costs if changes occur later.