Discussion 3: Agile

Discussion Topic:

How does agile change the scoping and sequencing of a project?
How does agile influence projects differently from a regular project?
You can use your project or some other project you have familiarity with.
If you have not worked on a project using agile, use a hypothetical project for your discussion question.

My Post:

Hello Class,
In the context of program management, the Agile model handles scope and sequencing very differently from traditional models such as the Waterfall model, which has a sequential linear approach to project management. On the other hand, Agile has an incremental, iterative, a cyclical approach to program management.
To answer this decision forum question, I am using my portfolio project for this class called Project Codebase. The project goal is to develop and implement an AI Agent and RAG (Retrieval-Augmented Generation) system to assist software engineering within a software development workflow.

How does agile change the scoping and sequencing of a project?

For regular projects, often the scope is defined mostly at the beginning of the project; this approach usually creates a Scope Baseline, which is an approved version of the scope of a project (Ucertify, n.d.). The Scope Baseline is a static document (cannot be modified), and it is based on the expectation that the project requirements, once gathered and defined, are unlikely to change during the following phases of the project management process, and stakeholder feedback is not needed until the end of the project. This approach to scope setting happens only within the Initial phase of the Waterfall model. For complex projects, such as Project Codebase is almost impossible to gather all requirements at the beginning of the project due to their complexity. In other words, the traditional approach to scope setting is too static to work well for Project Codebase; the project needs an approach that allows new or hidden requirements to be discovered during the development process, such as new features or modeling errors needing to be implemented or fixed. Moreover, stakeholders are not involved during the development; consequently, no feedback is gathered until deployment. On the other hand, the Agile model has a dynamic approach to scope setting, which fits well with the evolving nature of Project Codebase or any software development project. The model uses tools such as Product Backlog and User Stories to continuously and incrementally gather requirements and feedback from stakeholders, allowing the project scope to evolve with project development. This approach prioritizes value over process and changes over rigid adherence to a plan.

Regular projects usually have a sequential approach to project management, where phase A must finish before phase B starts. This sequential approach to program management does not implement incremental change, as each phase must be fully completed in one run; additionally, it does not allow retroactive changes to be made once the phase is finished. This approach is too rigid for a complex project such as Project Codebase. Project Codebase needs a more flexible

approach that breaks down complex problems into smaller problems and allows retroactive changes to be made; for example, modifying the RAG retrieval logic after testing reveals accuracy issues. On the other hand, the Agile model uses a cyclical phase system that implements weekly sprint/iteration sessions, making it better to address the high development and implementation uncertainty that comes with complex projects such as Project Codebase.

How does agile influence projects differently from a regular project?

As shown previously, using the Project Codebase, Agile influences projects by shifting the focus from strict adherence to a sequential process with low stakeholder involvement to prioritizing project value. It achieves this by implementing a cyclical approach to project management that involves stakeholders in the design, development, and implementation of the project, breaks complex problems into manageable pieces (increments) using plan-design-implementation sections (sprints), and adapts to change.

-Alex

References:

Ucertify (n.d.). Lesson 7: Project Schedule. Project Manager Professional (PMP) Based on PMBOK7. Ucertify. ISBN: 978-1-64459-415-5