Name: Shravani Sandesh Kulkarni
Student Id- 40272412

EXERCISE - CHAPTER 4

4.1 Find out all the reasons why risk management in the iterative development models is different compared to the traditional waterfall m

-> Risks are inevitable in any software development lifecycle. Regardless of the specifics of the projects and whether we are using a Waterfall or an Agile method, the team is sure to face some unforeseen or less expected issues during development. Preparing for and managing risks is essential in ensuring the project's success and avoiding overhead.

In a Waterfall scenario, where we'll be trying to plan risks well ahead of time, our estimation of the likelihood or severity of risks can be inaccurate. With constant changes in requirements in a shifting business environment, it is vital to define roles and responsibilities for continuously monitoring and controlling risks in our Waterfall project. Overlapping stages of development, poor quality assurance, and long processes are all sources of risk in a Waterfall environment.

With Agile projects, risk management usually encompasses a shorter timeframe. The risk register is reevaluated at every sprint planning meeting, and risks are discussed during stand-up meetings every day, and during retrospective meetings at the end of each iteration. Simply by using Agile, we can reduce a **variety of risks** related to budget, time to market, scope creep, requirements and security. However, we still need to define a process to manage risks.

Risk management in Agile takes on a more active and reactive role which is important to factor into daily activities. Waterfall typically has more time to plan or has longer projects or more stable environments, where requirements don't change that often. In traditional project management, the approach to Risk management is important, but it is not as active or ever-present as found in an agile environment.

-Here are the reasons why risk management in the iterative development model is different compared to the traditional waterfall model:

1) **Iterative nature:** Multiple cycles, or iterations, of development, take place in iterative models, and each iteration results in a functional increment of the product. Compared to the waterfall model, where risks are usually addressed later in the project lifetime, this iterative approach enables faster risk detection and mitigation.

2) **Risk Prioritization:** In iterative development, risks are frequently ranked according to how they might affect the current sprint or iteration. Risks that have the potential to impact the product's current iteration are prioritized higher, making sure that the most serious issues are resolved first.

3) **Regular Feedback Loops:** Iterative approaches place a strong emphasis on receiving regular input from end users and stakeholders. The team can swiftly adjust to changing requirements and identify possible risks early owing to this continuous feedback loop, which lessens the impact of risks on the project.

4) **Delivery of Workable Increments:** The goal of iterative models is to provide usable product increments after each iteration. By giving stakeholders early access to development feedback and enabling them to observe real progress, this incremental delivery method lowers the risk of project failure.

5) **Collaborative Approach:** Close cooperation between team members, stakeholders, and end users is encouraged by iterative development. By encouraging openness and transparency, this cooperative approach improves risk management by helping the team identify and address risks more successfully.

6) **Flexibility**: When compared to the waterfall approach, Iterative models are intrinsically more flexible. Teams can immediately address new risks by modifying the project's scope, or resources as necessary thanks to their adaptability.

7) **Continuous growth and development:** Regular retrospectives and reviews are a key component of iterative models, which promote continuous development. Through reflection on past performance and

identification of areas for development, these activities offer the opportunity to improve the risk management process for subsequent iterations.

8) **Flexibility in Requirements:** Iterative models provide flexibility in requirements, which helps the team early in the development process to prioritize and deliver high-value features. This adaptability lowers the chance of providing a product that falls short of stakeholders' requirements or expectations.

9) **Shorter timescales:** Iterative models frequently operate with frequent delivery cycles and shorter timescales. This shortened period shortens the amount of time that people are exposed to risks and enables prompt resolution of new problems.

10) **Emphasis on Testing:** Unlike waterfall models, which postpone testing until the very end, iterative models usually incorporate testing at every stage of development. Early and ongoing testing assists in identifying and reducing risks associated with performance, functionality, and quality.

In conclusion, because iterative development models emphasize continuous improvement, have frequent feedback loops, are flexible in requirements, have shorter timeframes, are collaborative, and are incremental in delivery, risk management in these models differs from the traditional waterfall model. These variations allow iterative models to efficiently recognize, rank, and reduce risks during a project.

References:

1) https://pmi-portland.org/news-and-content/675-risk-management-agile-v-waterfall

2) https://pm.stackexchange.com/questions/5957/how-is-risk-management-in-agile-development-different-from-risk-management-in-wa